

Estimating Volumes of Near-Spherical Molded Artifacts

Volume 115

Number 3

May-June 2010

David E. Gilsinn

Mathematical and Computational
Sciences Division, Information
Technology Laboratory,
National Institute of Standards
and Technology,
Gaithersburg, MD 20899-8910

Bruce R. Borchardt

Precision Engineering Division,
Manufacturing Engineering
Laboratory,
National Institute of Standards
and Technology,
Gaithersburg, MD 20899-8211

and

Amelia Tebbe

Department of Mathematics,
St. Mary's College of Maryland,
St. Mary's City, MD 20686-3001

david.gilsinn@nist.gov
bruce.borchardt@nist.gov
antebbe@smcm.edu

The Food and Drug Administration (FDA) is conducting research on developing reference lung cancer lesions, called phantoms, to test computed tomography (CT) scanners and their software. FDA loaned two semi-spherical phantoms to the National Institute of Standards and Technology (NIST), called Green and Pink, and asked to have the phantoms' volumes estimated. This report describes in detail both the metrology and computational methods used to estimate the phantoms' volumes. Three sets of coordinate measuring machine (CMM) measured data were produced. One set of data involved reference surface data measurements of a known calibrated metal sphere. The other two sets were measurements of the two FDA phantoms at two densities, called the coarse set and the dense set. Two computational approaches were applied to the data. In the first approach spherical models were fit to the calibrated sphere data and to the phantom data. The second approach was to model the data points on the boundaries of the spheres with surface B-splines and then use the Divergence Theorem to estimate the volumes. Fitting a B-spline model to the calibrated sphere data was done as a reference check on the algorithm performance. It gave assurance that the volumes estimated for the phantoms would

be meaningful. The results for the coarse and dense data sets tended to predict the volumes as expected and the results did show that the Green phantom was very near spherical. This was confirmed by both computational methods. The spherical model did not fit the Pink phantom as well and the B-spline approach provided a better estimate of the volume in that case.

Key words: B-splines; computed tomography; coordinate measuring machine; divergence theorem; lung cancer; lung cancer phantoms; nonlinear least squares.

Accepted: December 28, 2009

Available online: <http://www.nist.gov/jres>

1. Introduction

Tomography is a method of imaging a single slice of the body. Modern computed tomography (CT) is a medical imaging method that uses tomography, but also employs digital image processing techniques, to generate three dimensional images built from a large sequence of two-dimensional x-ray images made around a single axis. CT has shown promising results in detecting lung cancers at more operable stages, when survival is better [1].

The Food and Drug Administration (FDA) is conducting research on developing reference cancer

lesions, called phantoms, to test CTs and their software. Two samples were loaned to NIST to estimate volumes. The material of the phantoms simulates lung cancer material. The phantoms can be inserted into a simulated body torso for CT scans. The two phantoms are shown in Fig. 1. Although they seem spherical, they are slightly non-spherical. The larger one on the right is referred to as the Green phantom and the one on the left is called the Pink phantom due to the material colors.

One experimental approach to estimate the volume of the phantoms would be to use an Archimedes test in which the phantoms would be immersed in a liquid bath in a well calibrated container with fine measurement



Fig. 1. Two simulated lung cancers, called phantoms.

gradations to determine liquid displacement. However, in the case of these phantoms, the material used to manufacture these phantoms was porous and the phantoms would have to be coated. This, of course, would affect the “ground truth” volume estimate. As a result, the approach chosen for this study was based on a fundamental theorem in calculus, called the Divergence Theorem (see Taylor [2]), an analogue of Green’s Theorem in two dimensional space. In the Divergence Theorem a volume integral is shown to be equal to a surface integral. Therefore, we surmised that, if a model of the surfaces of the phantoms could be developed, the Divergence Theorem would help to estimate their volumes. In order to develop a surface model we needed to obtain data about the surfaces of the phantoms. This was done using a coordinate measuring machine (CMM) in the Manufacturing Engineering Laboratory (MEL) at NIST. This machine produced a set of (x, y, z) points on the surface of each phantom. The data were then transformed to spherical coordinates and modeled using a set of basis functions, called B-splines. After fitting, the B-spline model was employed to generate a grid of values on the surface. These values were used to form surface triangles that were then used to compute the necessary surface integrals and finally the volume. The quality of the volume estimates depended on the surface grid sizes, as will be clear from the discussion below. The method of B-spline surface modeling is not new, in that it was suggested in the book by Dierckx [3]. However the application in the current case, that uses the Divergence Theorem, appears to be new. A reader can also consult the Dierckx references [4] and [5].

The paper is divided as follows. Section 2 describes how a volume can be computed using the Divergence

Theorem. Section 3 describes the surface point generation experiments using the CMM. Section 4 introduces the two methods of data modeling used to estimate the phantoms’ volumes. The first uses a spherical model. The second uses the B-splines as a basis for a least squares fit with regularization to model the phantoms’ surface data. The fitted functions were then used to generate the grid data necessary to apply the Divergence Theorem. Section 5 presents the specific surface triangulation used to implement the Divergence Theorem. Section 6 describes the volumetric results from applying the spherical and B-spline models to the CMM data. A summary is given in Sec. 7.

2. Volume Estimation by the Divergence Theorem

In this section we state the Divergence Theorem and indicate how it can be used to estimate the volume of a polyhedron. This provides the motivation for the need to determine surface data points from the phantoms and then to build a surface model that is used to create data points at surface triangulation vertices. As the number of triangles was increased, the volume estimates were expected to tend to a fixed value. As will be seen, this expectation is confirmed below.

2.1 Divergence Theorem in 3-D and Volume Computation

Let Ω be a simply connected region in three dimensional space and Γ the surface boundary. Then

$$\iiint_{\Omega} \vec{\nabla} \cdot F \, dx dy dz = \iint_{\Gamma} F \cdot \hat{n} \, d\sigma, \quad (1)$$

where $F(x, y, z) = (F_1(x, y, z), F_2(x, y, z), F_3(x, y, z))$ is a differentiable vector field,

$$\vec{\nabla} \cdot F = \frac{\partial F_1}{\partial x} + \frac{\partial F_2}{\partial y} + \frac{\partial F_3}{\partial z}, \quad (2)$$

is the divergence of the vector field, \hat{n} is an outward-pointing unit normal vector to Γ , and $d\sigma$ is an infinitesimal element of the surface.

If we select $F(x, y, z) = (1/3)(x, y, z)$, then $\vec{\nabla} \cdot F = 1$ and we can write

$$\iiint_{\Omega} dx dy dz = \frac{1}{3} \iint_{\Gamma} \hat{n} \cdot (x, y, z) \, d\sigma. \quad (3)$$

Note that the left hand side is simply the volume of the region Ω . The $1/3$ factor comes from the definition of $F(x, y, z)$ so that $\vec{\nabla} \cdot F = 1$. Now, if Γ is approximated by disjoint polyhedra (planar surface patches), Γ_i , then

$$\Gamma = \bigcup_{i=1}^N \Gamma_i, \quad (4)$$

and

$$\frac{1}{3} \int_{\Gamma} \hat{n} \cdot (x, y, z) d\sigma = \frac{1}{3} \sum_{i=1}^N \int_{\Gamma_i} \hat{n}_i \cdot (x, y, z) d\sigma, \quad (5)$$

where \hat{n}_i is the unit outer normal to Γ_i . Here we will model the surface patches by planar facets and, in particular, triangular facets. The plane for Γ_i is given by $\hat{n}_i \cdot (x, y, z) = c_i$, where c_i is a constant associated with each triangular facet. The sum of the integrals over the facets thus reduces to

$$\frac{1}{3} \sum_{i=1}^N \int_{\Gamma_i} \hat{n}_i \cdot (x, y, z) d\sigma = \frac{1}{3} \sum_{i=1}^N \int_{\Gamma_i} c_i d\sigma = \frac{1}{3} \sum_{i=1}^N c_i \text{Area}(\Gamma_i). \quad (6)$$

This method of computing the volume of an object from a surface integral can be found in Schneider and Eberly [6].

2.2 Area of a Planar Polyhedron in 3-D

In order to estimate the surface area of a facet, as used in (6), we will describe the process of computing the area of a planar polyhedron in space by use of Stokes' Theorem and then we will particularize it to a planar triangle in space. Here the assumption is made that there is surface data available at the planar triangle vertices. Stokes' Theorem states that if C is a piecewise smooth boundary curve, oriented positively, of a surface Γ , and if F is a differentiable vector field defined on Γ and \hat{n} is a unit normal satisfying the right-hand rule relative to the boundary orientation, then

$$\int_{\Gamma} (\vec{\nabla} \times F) \cdot \hat{n} d\sigma = \int_C F \cdot d\vec{R}, \quad (7)$$

where the curl of $F(x, y, z) = (F_1(x, y, z), F_2(x, y, z), F_3(x, y, z))$ is

$$\vec{\nabla} \times F = \left(\frac{\partial F_3}{\partial y} - \frac{\partial F_2}{\partial z}, \frac{\partial F_1}{\partial z} - \frac{\partial F_3}{\partial x}, \frac{\partial F_2}{\partial x} - \frac{\partial F_1}{\partial y} \right), \quad (8)$$

and $d\vec{R} = (dx, dy, dz)$. If we set $F(x, y, z) = (1/2) \hat{n} \times (x, y, z)$ then $\vec{\nabla} \times F = \hat{n}$ and $\hat{n} \cdot \hat{n} = 1$ so that

$$\begin{aligned} \text{Area}(\Gamma) &= \int_{\Gamma} d\sigma = \frac{1}{2} \oint_C (\hat{n} \times (x, y, z)) \cdot (dx, dy, dz), \\ &= \frac{1}{2} \oint_C \hat{n} \cdot ((x, y, z) \times (dx, dy, dz)), \\ &= \frac{1}{2} \oint_C \hat{n} \cdot ((x(t), y(t), z(t)) \times \\ &\quad (x'(t), y'(t), z'(t))) dt, \\ &= \frac{1}{2} \oint_C \hat{n} \cdot (y(t)z'(t) - z(t)y'(t), \\ &\quad z(t)x'(t) - x(t)z'(t), \\ &\quad x(t)y'(t) - y(t)x'(t)) dt. \end{aligned} \quad (9)$$

The factor $1/2$ comes from the definition of $F(x, y, z)$ so that $(\vec{\nabla} \times F) = \hat{n}$.

We will now specialize the Stokes formula to find the area of a spatial triangle, γ , in terms of its three vertices oriented positively. Let the three vertices, in positive orientation, be specified by $v_1 = (x_1, y_1, z_1)$, $v_2 = (x_2, y_2, z_2)$, $v_3 = (x_3, y_3, z_3)$. Parameterize the boundary of the triangle as follows. For $t \in [0, 1]$, let $v = v_1 + t(v_2 - v_1) = (x_1 + t(x_2 - x_1), y_1 + t(y_2 - y_1), z_1 + t(z_2 - z_1))$. For $t \in [1, 2]$ let $v = v_2 + (t - 1)(v_3 - v_2) = (x_2 + (t - 1)(x_3 - x_2), y_2 + (t - 1)(y_3 - y_2), z_2 + (t - 1)(z_3 - z_2))$. Finally, for $t \in [2, 3]$, put $v = v_3 + (t - 2)(v_1 - v_3) = (x_3 + (t - 2)(x_1 - x_3), y_3 + (t - 2)(y_1 - y_3), z_3 + (t - 2)(z_1 - z_3))$. The area of the spatial triangle, in terms of the parameterized boundary, can be written as

$$\begin{aligned} \text{Area}(\gamma) &= \frac{1}{2} \oint_C \hat{n} \cdot (y(t)z'(t) - z(t)y'(t), \\ &\quad z(t)x'(t) - x(t)z'(t), \\ &\quad x(t)y'(t) - y(t)x'(t)) dt, \\ &= \frac{1}{2} \hat{n} \cdot \left(\int_0^3 (y(t)z'(t) - z(t)y'(t)) dt, \right. \\ &\quad \int_0^3 (z(t)x'(t) - x(t)z'(t)) dt, \\ &\quad \left. \int_0^3 (x(t)y'(t) - y(t)x'(t)) dt \right) \end{aligned} \quad (10)$$

By straightforward integration over each of the parameterized segments it is easy to show that

$$\begin{aligned} \text{Area}(\gamma) = & \frac{1}{2} \hat{n} \cdot ((y_1 z_2 - y_2 z_1) + (y_2 z_3 - y_3 z_2) + (y_3 z_1 - y_1 z_3), \\ & (z_1 x_2 - z_2 x_1) + (z_2 x_3 - z_3 x_2) + (z_3 x_1 - z_1 x_3), \\ & (x_1 y_2 - x_2 y_1) + (x_2 y_3 - x_3 y_2) + (x_3 y_1 - x_1 y_3)). \end{aligned} \quad (11)$$

The normal vector to the oriented triangle can be computed as follows. Let $\vec{v} = (x_2 - x_1, y_2 - y_1, z_2 - z_1)$ and $\vec{w} = (x_3 - x_1, y_3 - y_1, z_3 - z_1)$. Then $\vec{n} = \vec{v} \times \vec{w} = ((y_2 - y_1)(z_3 - z_1) - (z_2 - z_1)(y_3 - y_1), (z_2 - z_1)(x_3 - x_1) - (x_2 - x_1)(z_3 - z_1), (x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1))$. If we set $n_1 = (y_2 - y_1)(z_3 - z_1) - (z_2 - z_1)(y_3 - y_1)$, $n_2 = (z_2 - z_1)(x_3 - x_1) - (x_2 - x_1)(z_3 - z_1)$, $n_3 = (x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1)$, then $\|\vec{n}\| = \sqrt{n_1^2 + n_2^2 + n_3^2}$ and $\hat{n} = \vec{n} / \|\vec{n}\|$.

3. Surface Metrology of the Artifacts

In this section we discuss the experimental method used to obtain surface data for the three objects used in this study. As a test object for the modeling process, a well calibrated sphere was selected. This object, along with the FDA phantoms formed the study artifact set. Data points, in the form of (x, y, z) coordinates, were created by the probing action of the CMM. Figure 2 shows the CMM system used to measure the artifacts. The system is computer controlled and touches an object to be measured at programmed points in order to produce (x, y, z) values at the probed points.

Figure 3 shows one of the phantoms, held by a device called a vacuum chuck, as it is being touched by the CMM probe. The probe itself can be programmed to approach an object at various angles. In the background of the figure is a high quality reference steel sphere. Before an object is probed the reference sphere is measured to determine the effective diameter of the CMM probe tip. This reference sphere is separate from the calibrated sphere used to test the modeling process. The difference between the known diameter of the reference steel sphere and the apparent diameter of the measured reference metal sphere gave the calibrated effective probe diameter.



Fig. 2. Computer controlled CMM.



Fig. 3. Lung nodule phantom on the vacuum chuck.

The FDA phantoms were created by a molding process and the mold marks on both spheres were visible. These mold marks were used to align the phantoms with the coordinate system of the CMM. The marks were laid out like lines of latitude, leading to the use of a natural nomenclature of latitude and longitude like those of the Earth. For each phantom, the North Pole was chosen to be the one with darker, deeper, or more obvious mold marks.

For the purpose of understanding the measurement process we will describe the physical fixture positioning of the phantoms. They were held by a vacuum chuck (see Fig. 3) to minimize distortion and reduce the chance of damaging the spheres' surfaces. As Fig. 3 shows, the chuck has a shallow cone to hold the phantoms. The phantoms contacted the cone around a circle of latitude at about -45° . This was measured from the equatorial circle around the middle of the phantoms. For example, the point at the top of the phantoms would be at $+90^\circ$. The wall vacuum of the cone was strong enough to hold the spheres sufficiently that they did not move significantly, as shown by the repeatable results from run to run at the $1\text{ }\mu\text{m}$ level. The setup of the phantoms in the vacuum chuck was accomplished by eye alignment using the visible mold marks and minor imperfections as guides to the eye. The expanded uncertainty of alignment was estimated to be approximately $\pm 2^\circ$ for the vertical angle (i.e., keeping the equator horizontal) and $\pm 4^\circ$ for the azimuthal angle. For a presentation of the guidelines for how expanded uncertainty of parameters is computed see Taylor and Kuyatt [7]. Essentially, the guideline for estimating the expanded uncertainty involves applying a multiplier, $k = 2$, times the square root of the variance about a predicted parameter value (i.e., two times the standard error). For a full discussion of confidence intervals for parameter estimation see Draper and Smith [8].

Three sets of measurements were planned for each of the phantoms. In each, points were probed on only a hemisphere at a time and later the data sets were post-processed to form a spherical data set. In the first set of measurements the North Pole was set as the top point. A set of points was programmed to probe the top hemisphere down to the equator. The phantom was then re-positioned in the chuck so that the South pole was the top point. The same hemispherical points were probed. The rotation was done in such a manner that the mold marks representing the longitudinal lines were kept as aligned as possible. Post-processing of the data associated the correct signs with the measured coordinates relative to the CMM coordinate system. The third measurement involved re-positioning the phantoms so

that the equatorial circle was vertical and the North-South axis was horizontal. Again two hemisphere sets of probe points were measured. This position was not feasible for the Pink phantom in one of the experiments described below.

Visually, the pink sphere was noticeably out-of-round, in the shape of an oblate spheroid. A dial caliper gave diameter measurements given in Table 1. The uncertainty of caliper measurements on hard steel surfaces is about 0.1 mm , and is estimated to be about 0.3 mm on the sample spheres due to the potential that the contact force would distort the soft surfaces of the spheres. All estimated uncertainties were $k = 2$ expanded uncertainties.

Table 1. Dial Caliper Measurements of Phantoms' Diameters

	Equatorial	Polar
Pink	20.0 mm	18.4 mm
Green	20.0 mm	20.1 mm

Two probing experiments were performed on each of the phantoms and the calibrated sphere. They created what we will call a coarse data set and a dense data set. For the coarse data set the plan was to measure each phantom on the CMM three times in each position, with 61 coordinate points per hemisphere. For the green sphere, each measurement set consisted of three separate sets of points: North pole up, South pole up, and prime meridian/equator intersection up. The pink sphere could not be held sideways in the first experiment, as the out-of-roundness prevented an effective vacuum seal. Therefore, a measurement data set for this sphere had only the North Pole up and South Pole up data. The third data set in this case was a re-measure of the North Pole up position. Each data set consisted of 122 points. The plots in Fig. 4 and 5 show the radial deviation from a best-fit sphere for the full data sets. The figures show the radial residuals obtained by fitting sphere models to the Green and Pink data with an algorithm ordinarily used during sphere calibration work. In particular, they represent the residual errors between the distance from the fitted sphere center to the probed points and the fitted radius of the sphere model. The residuals, in the case of the Green phantom, range from approximately -0.1 mm to $+0.1\text{ mm}$, whereas the residuals, in the case of the Pink phantom, range from approximately -0.57 mm to $+0.23\text{ mm}$. This suggests the slight non-spherical nature of the Pink phantom. Figure 6 shows the typical distribution of the probe points on a sphere. The plot is a transparency so

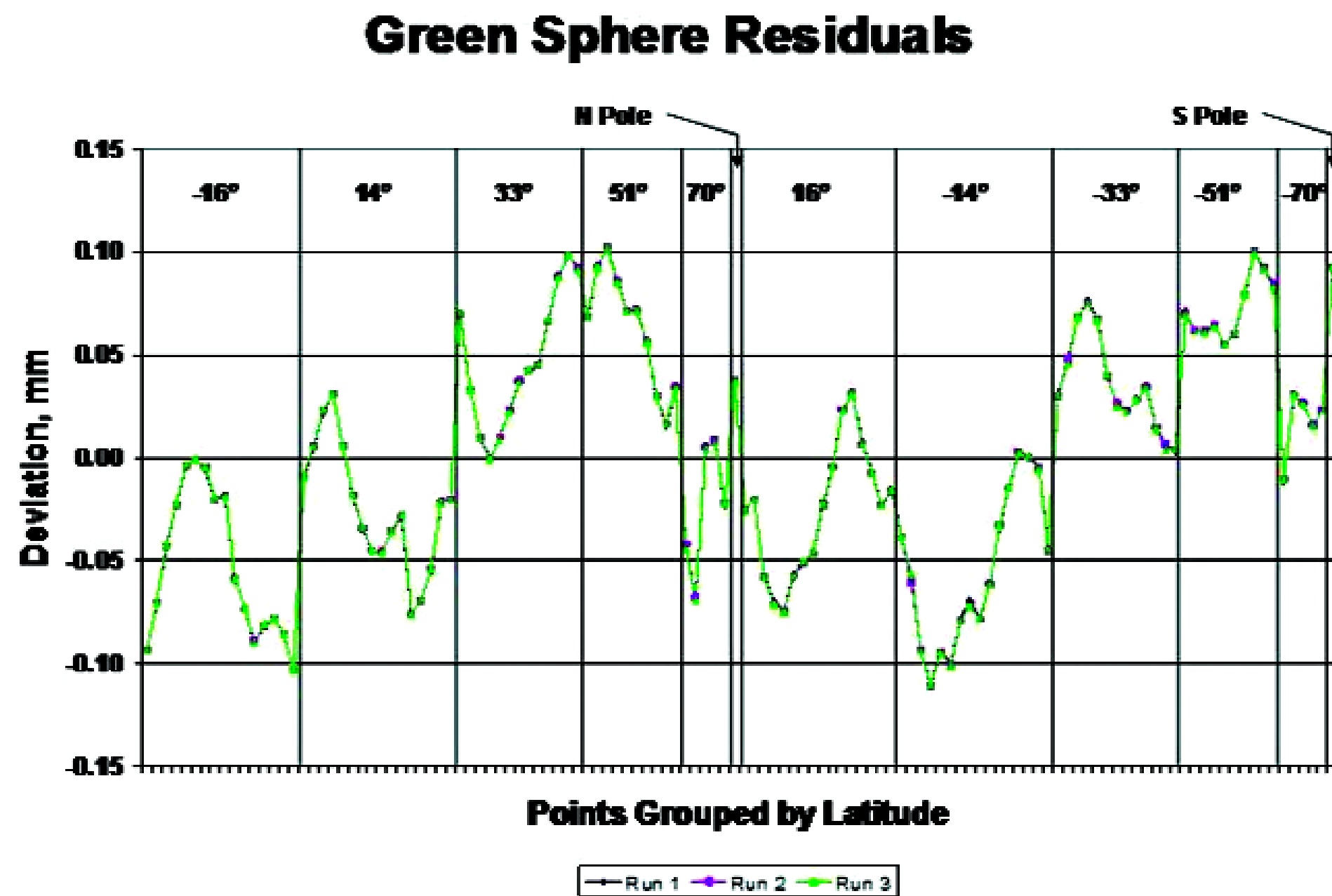


Fig. 4. Residual Measures CMM Errors for the Green Phantom Sphere.

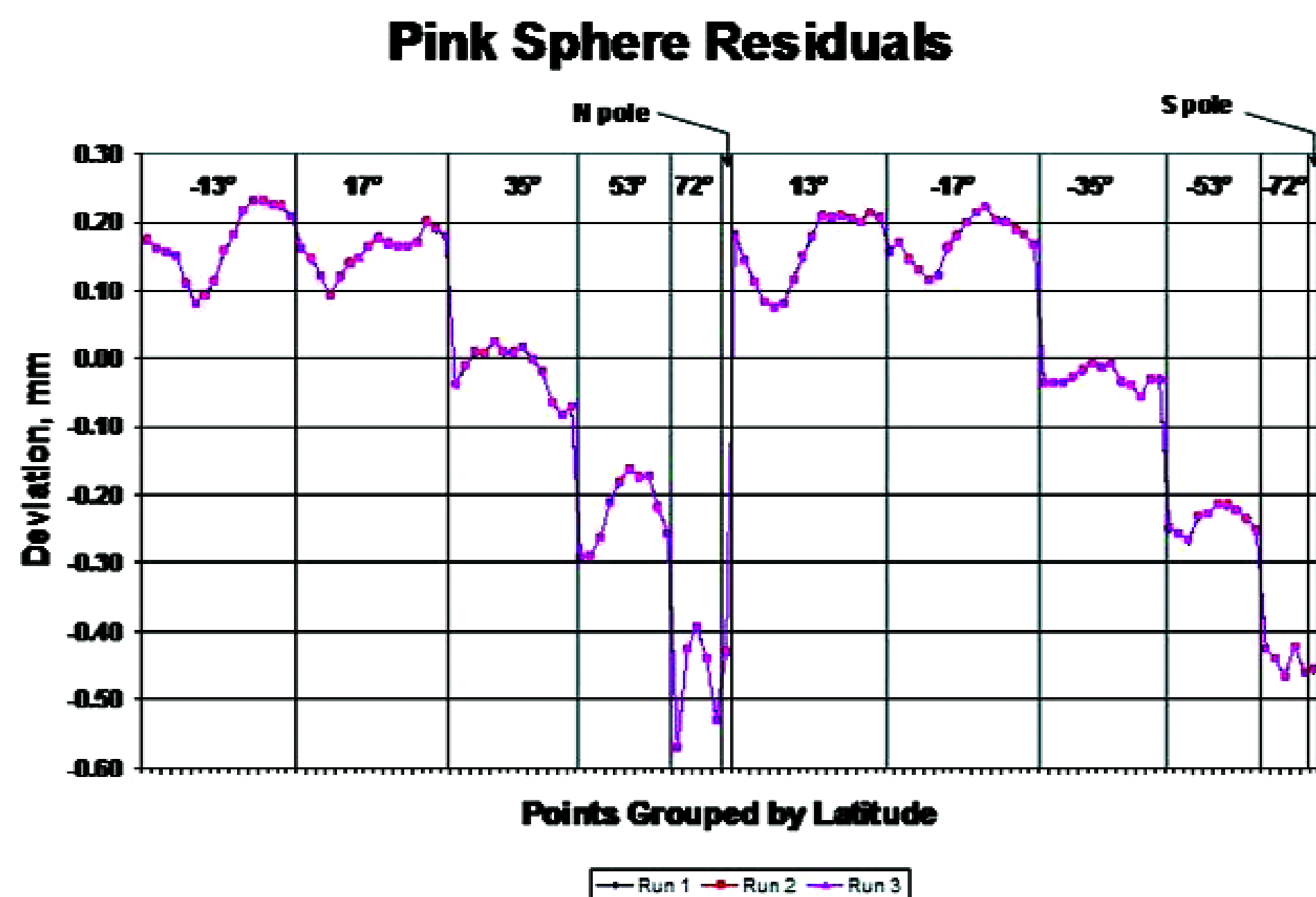


Fig. 5. Residual Measures CMM Errors for the Pink Phantom Sphere.

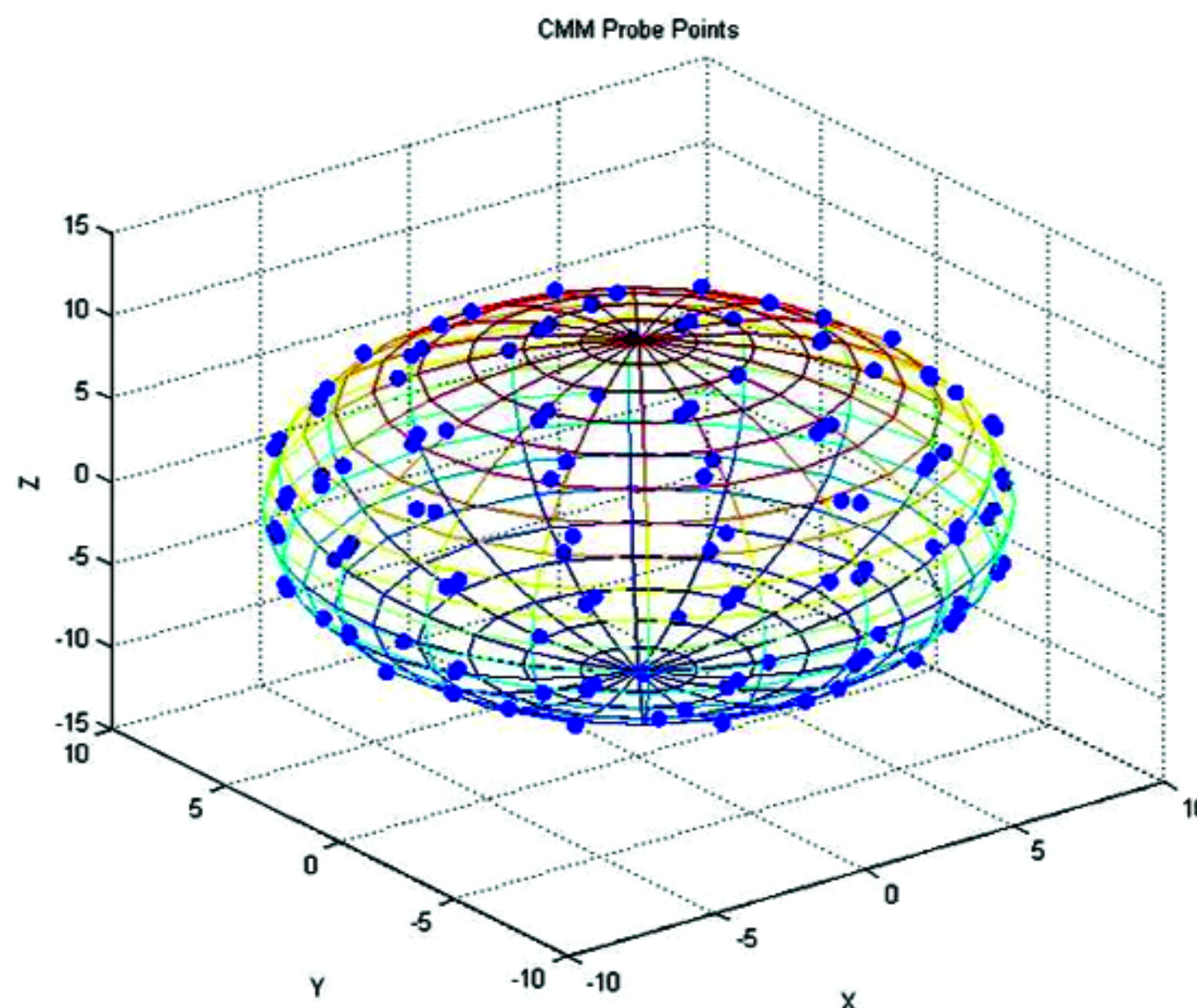


Fig. 6. Sample Distribution of the CMM Probe Points on a Sphere.

that probe points on the opposite side of the sphere are visible. The calibrated sphere on the shaft, with a diameter of 19.05 mm, was also measured with 122 points, repeated five times. It was measured in one position, since it was permanently mounted on a support shaft.

For the dense data set 181 points were taken on the phantoms and the calibrated sphere, with five repeats in each position. The positions were taken the same as those in the first experiment. That is, the alignment was selected with North pole up (position 1), South pole up (position 2), and prime meridian/equator intersection up (position 3). In this case it was possible to hold the Pink phantom in the sideways position 3. The calibrated sphere was also measured at 181 points with five repeats.

4. Modeling Methodologies

In this section two forms of data modeling will be discussed. Since the phantoms seemed to be nearly spherical the natural tendency was first to consider fitting a spherical model to each phantom and estimating the volume of the fitted spheres. However, in order to develop a potentially more accurate volume estimation model, the surface data was also fit using tensor products of B-splines and the volumes estimated by the Divergence Theorem.

4.1 A Spherical Model

In this section we consider how close the data could be modeled by a spherical model for the data. The calibrated sphere, of course, could be modeled via a spherical model.

In particular, let $c = (c_1, c_2, c_3, c_4)$ and set

$$F(x, y, z, c) = (x - c_1)^2 + (y - c_2)^2 + (z - c_3)^2 - c_4^2. \quad (12)$$

The unknown parameters c_1, c_2, c_3 represent the center of the sphere and c_4 is the radius. All of the data were measured in millimeters so that the parameters naturally have millimeter units. Define

$$O(c) = \sum_{i=1}^n \left\{ (x_i - c_1)^2 + (y_i - c_2)^2 + (z_i - c_3)^2 - c_4^2 \right\}^2, \quad (13)$$

where O , the objective function, is a measure of the residual for the fitted sphere defined by the coefficients c . Since the function O is a nonlinear implicit function of the parameters we needed to use a nonlinear minimization algorithm to find the best fit, i.e., to solve the problem

$$\min_c O(c). \quad (14)$$

There are a number of algorithms for fitting least squares models to data on geometric shapes (see Shakarji [9]). There are also various algorithms for minimizing general nonlinear functions, such as (13). All of the algorithms involve iterative minimization of some form. Many require computing derivatives of the objective function in order to generate search directions along which to identify a minimum. Others do not involve derivatives but may be somewhat slower in the minimization search. The algorithm selected here, because of the relatively few parameters involved, and the fact that derivatives are not required, is a form of polyhedron search method called the Nelder-Mead method (see Sauer [10]). The Newton method, requiring derivatives, was initially used to estimate the parameters, but the Nelder-Mead tended to produce the smallest value to (13).

The Nelder-Mead algorithm works iteratively through steps that involve reflections, vertex extensions, and multidimensional polyhedron contractions until the volume of the polyhedron becomes less than a prescribed tolerance. The polyhedron vertices then enclose the minimum. The median value of the polyhedron vertex values is taken as the minimum value.

The uncertainties of the estimated center and radius were computed using the methods proposed in Draper and Smith [8] for nonlinear regression. In particular, if $\hat{c} = (\hat{c}_1, \hat{c}_2, \hat{c}_3, \hat{c}_4)$, then define the $n \times 4$ matrix with elements

$$\hat{Z}_{ij} = \frac{\partial F}{\partial c_i}(x_i, y_i, z_i, \hat{c}), \quad i=1, \dots, n \quad j=1, 2, 3, 4. \quad (15)$$

The i -th row is given by

$$\hat{Z}_i = [-2(x_i - \hat{c}_1) \quad -2(y_i - \hat{c}_2) \quad -2(z_i - \hat{c}_3) \quad -2\hat{c}_4]. \quad (16)$$

The standard error of \hat{c}_i , $s.e.(\hat{c}_i)$, is given by the square root of the i -th diagonal element of $(\hat{Z}'\hat{Z})^{-1}s^2$ where $s = O(\hat{c}) / (n - 4)$. The expanded uncertainty is given by $2s.e.(\hat{c}_i)$ as defined in Taylor and Kuyatt [20]. The uncertainty limits of \hat{c}_i are $\hat{c}_i \pm 2s.e.(\hat{c}_i)$. All units are in millimeters except volume, which is in cubic millimeters.

4.2 A B-spline Model

Given measured data points on the surfaces of the calibrated sphere and the two phantoms, surface models can be constructed using B-splines as basis functions. In this section we will define cubic B-splines and show the construction of tensor products of B-splines.

4.2.1 Cubic B-Splines in One Variable

Suppose that a function $y = f(x)$, $x \in [a, b]$, is known at the n points $(x_1, y_1), \dots, (x_n, y_n)$, where $a < x_1 < x_2 < \dots < x_n < b$, $y_q = f(x_q)$, $q = 1, \dots, n$. a and b are finite interval bounds. It is known that a polynomial of degree $n - 1$, $P(x)$, can be constructed to pass through these n points. In the case of highly accurate data points this polynomial can be constructed to interpolate these n points by, for example, Lagrange polynomial or Newton divided difference algorithms. But, for large n , it is also well known that polynomials of high degree can produce unwanted oscillations between the interpolated points. It is crucial then to approximate sets of data with as low degree polynomials as possible. Of course these polynomials may or may not interpolate the data points but may be made to come as close to them as possible. The ability to create highly flexible approximants from low-degree polynomials is a significant advantage of functions called splines.

Given a set of \hat{r} real numbers, satisfying $a < \xi_1 < \xi_2 < \dots < \xi_{\hat{r}} < b$, a spline function, $F(x)$, of order p (or degree $p - 1$) with knots $\xi_1, \xi_2, \dots, \xi_{\hat{r}}$ is a function that satisfies two properties: (1) in each of the intervals $x \leq \xi_1$; $\xi_{j-1} \leq x \leq \xi_j$, ($j = 2, 3, \dots, \hat{r}$); $\xi_{\hat{r}} \leq x$, $F(x)$ is a polynomial of degree $p - 1$ or less. (2) $F(x)$ and its derivatives of orders $1, 2, \dots, p - 2$ are continuous. This would mean, for example, a spline function of order four would be constructed from polynomials of degree three (cubic) or less on the intervals $x \leq \xi_1$; $\xi_{j-1} \leq x \leq \xi_j$, ($j = 2, 3, \dots, \hat{r}$); $\xi_{\hat{r}} \leq x$ with continuous derivatives of orders one and two.

A well known mathematical technique to construct complicated functions is to form linear combinations of simpler functions, called basis functions. In the current paper, data sets will be approximated by linear combinations of special spline functions, called B-splines, for Basis splines. It is known that any spline function can be represented in terms of B-splines. This particular basis has the advantage of leading to computational algorithms that are elegant, efficient, and stable. Only B-splines of order four will be considered here. They are cubic splines that are non-zero only over four adjacent intervals between knots (see Fig. 7). The notation for a B-spline is $N_{p,i}(x)$, where $N_{p,i}(x)$, is zero everywhere except in the range $\xi_{i-p} < x < \xi_i$, where in this work $p = 4$. To simplify notation let $N_i(x) = N_{4,i}(x)$. Then a B-spline of order four, or cubic B-spline, is a cubic spline with knots $\xi_{i-4}, \xi_{i-3}, \xi_{i-2}, \xi_{i-1}, \xi_i$ that is zero everywhere except in the range $\xi_{i-4} < x < \xi_i$. $N_i(x)$ is defined uniquely except for a scaling factor and is conventionally taken to be

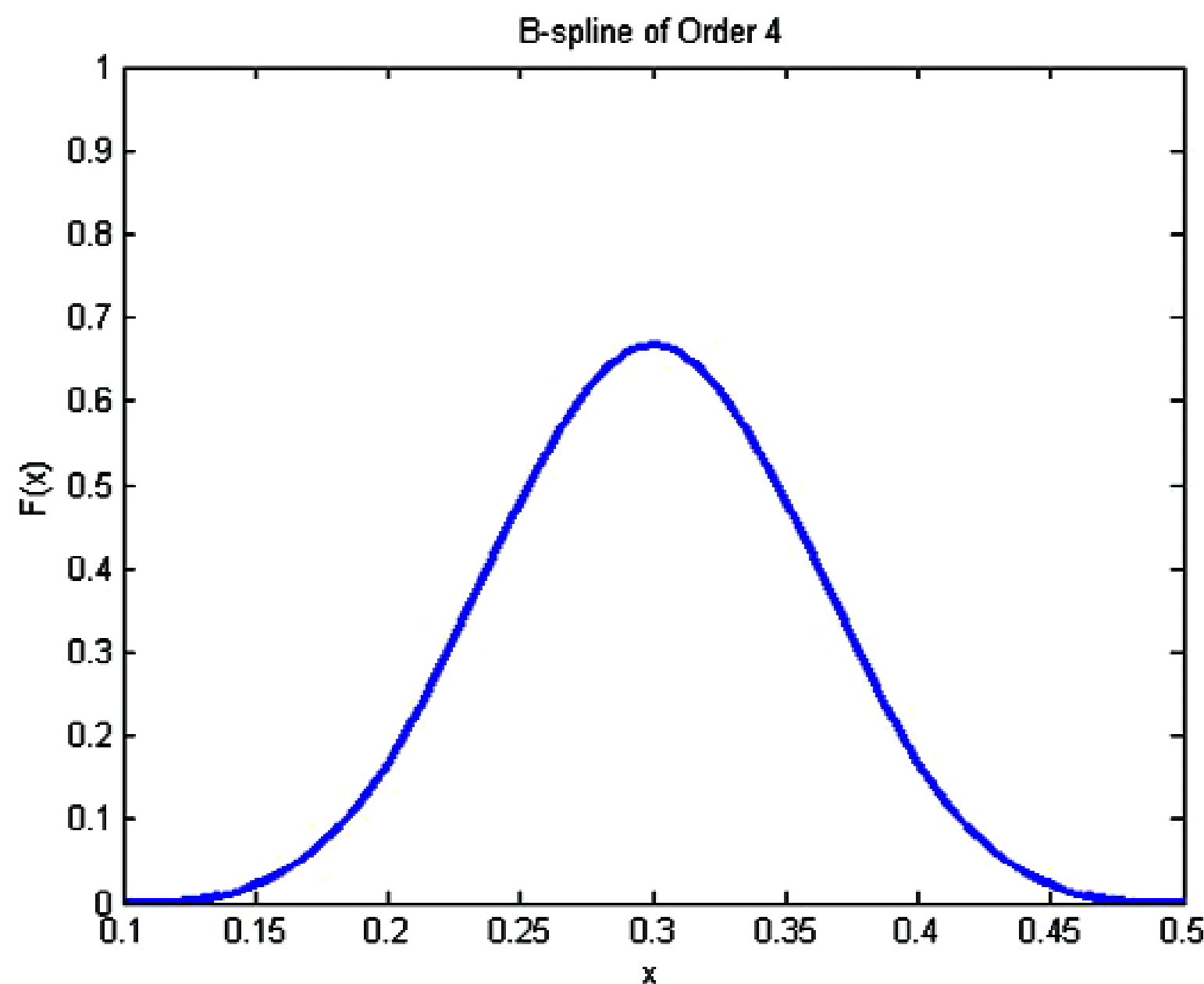


Fig. 7. B-spline of Order 4 with knots at $x = 0.1, 0.2, 0.3, 0.4, 0.5$.

positive throughout the range $\xi_{i-4} < x < \xi_i$ and has a single maximum value. Since $N_i(x)$ is a cubic spline it has continuous derivatives of order one and two at ξ_{i-4} and ξ_i . These derivatives are zero at the endpoint knots.

To define a complete set of B-splines on the set of points $a < \xi_1 < \xi_2 < \dots < \xi_{\hat{r}} < b$ it is necessary to introduce eight additional points at the boundaries given by $\xi_{-3}, \xi_{-2}, \xi_{-1}, \xi_0, \xi_{\hat{r}+1}, \xi_{\hat{r}+2}, \xi_{\hat{r}+3}, \xi_{\hat{r}+4}$. It is usual to have $\xi_0 = a, \xi_{\hat{r}+1} = b$. With this augmented set of knots one can define $\hat{r} + 4$ fundamental cubic B-splines, $N_i(x)$, $i = 1, 2, \dots, \hat{r} + 4$. Then the general cubic B-spline has a unique representation in the range $a \leq x \leq b$ of the form

$$F(x) = \sum_{i=1}^r c_i N_i(x), \quad (17)$$

where $r = \hat{r} + 4$.

There are computational advantages in using cubic B-splines. For any given x , all but four adjacent $N_i(x)$ are zero. In particular, if $x \in [\xi_{i-1}, \xi_i]$, the four non-zero cubic B-splines are $N_i(x), N_{i+1}(x), N_{i+2}(x), N_{i+3}(x)$.

A least squares curve fitting problem to the data $a < x_1 < x_2 < \dots < x_n < b, y_q = f(x_q), q = 1, \dots, n$, becomes one of determining the coefficients c_i as the least squares solution to the equations

$$\sum_{i=1}^r c_i N_i(x_q) \approx y_q, \quad q = 1, 2, \dots, n. \quad (18)$$

These may be written in matrix notation as

$$Ac \approx y, \quad (19)$$

where A is the $n \times r$ matrix whose element in column i of row q is $N_i(x_q)$ and c, y are column vectors with elements $c_i, y_q, q = 1, \dots, n$, respectively. If the data points are arranged in increasing order of x , then the matrix A becomes a banded matrix with bandwidth four. For a more thorough discussion of B-splines and their computation see de Boor [11] and Cox [12].

There are two functions in the MATLAB Spline Toolbox that implement the evaluation of B-splines. A set of knots can be augmented at the ends by the function **augknt** and the B-splines and their derivatives can be evaluated by the function **spcol**.

4.2.2 Tensor Products of Cubic B-Splines in Two Variables

In this section the B-spline concept will be extended to two dimensions in order to fit two dimensional scattered data by a surface function. In this surface-fitting problem data points $(x_q, y_q), q = 1, 2, \dots, n$, and values at these points, $z_q = f(x_q, y_q), q = 1, 2, \dots, n$, are given. The surface model used to fit these data points will involve sums of products of B-splines.

To introduce this model, define a rectangle, say R , by $a \leq x \leq b, c \leq y \leq d$ in the (x, y) plane. The definition

does not restrict the data points to the Euclidean plane. They could, for example be angular coordinates, as will be seen later. The rectangle is subdivided by sets of knots ξ_i , $i = 1, 2, \dots, \hat{r}$ and η_j , $j = 1, 2, \dots, \hat{s}$, where $a < \xi_1 < \xi_2 < \dots < \xi_{\hat{r}} < b$ and $c < \eta_1 < \eta_2 < \dots < \eta_{\hat{s}} < d$, where \hat{r} , \hat{s} are indices, not necessarily equal and a , b , c , d are the bounds on the rectangle R . The knots are then extended by eight in each dimension as done in the one dimensional case. These knots divide the rectangle R into rectangular panels in the plane given by R_{ij} , $i = 1, 2, \dots, \hat{r}$, $j = 1, 2, \dots, \hat{s}$. Then, a basis set of splines for this pairing of knots can be constructed as products of B-splines $N_i(x)N_j(y)$. In fact, the surface spline model is given by

$$F(x, y) = \sum_{i=1}^r \sum_{j=1}^s c_{ij} N_i(x) N_j(y), \quad (20)$$

called a tensor product of splines, where $r = \hat{r} + 4$ and $s = \hat{s} + 4$.

As in the one dimensional case, these tensor product splines have a number of advantages. First of all, the basis functions $N_i(x)N_j(y)$ are each non-zero over a rectangle composed of sixteen panels in a 4×4 arrangement. In particular $N_i(x)N_j(y)$ is non-zero only when $\xi_{i-4} \leq x \leq \xi_i$ and $\eta_{j-4} \leq y \leq \eta_j$. Next, if (x_q, y_q) , $q = 1, 2, \dots, n$ and values at these points, $z_q = f(x_q, y_q)$, $q = 1, 2, \dots, n$ are given, then the fitting problem can be formulated as finding the least squares solution of

$$\sum_{i=1}^r \sum_{j=1}^s c_{ij} N_i(x_q) N_j(y_q) \approx z_q, \quad q = 1, 2, \dots, n. \quad (21)$$

Again, this can be written in a matrix form as

$$Ac \approx z, \quad (22)$$

where A is now a matrix with n rows and rs columns, and z is a column vector with n rows. The elements in row q , $q = 1, 2, \dots, n$, of A are formed as follows. We start with a fixed j , $j = 1, 2, \dots, s$, and let i , $i = 1, 2, \dots, r$ vary for that j . Then, the element in column $(j-1)r + i$ for row q is given by $N_i(x_q)N_j(y_q)$. The j is then incremented and the i 's varied again. The end resulting column values in A for row q would look like $N_1(x_q)N_1(y_q) N_2(x_q)N_1(y_q) \dots N_r(x_q)N_1(y_q) N_1(x_q)N_2(y_q) \dots N_r(x_q)N_2(y_q) \dots N_1(x_q)N_s(y_q) \dots N_r(x_q)N_s(y_q)$. For a full discussion of tensor products of B-splines see de Boor [11], Eberly [13], and Rogers and Adams [14].

4.2.3 An Issue in Computing Tensor Product B-Splines and the Relation to Least Squares

Unfortunately, for some choices of knots the resulting matrix A in (22) might be rank deficient and it would not be possible to use the normal equations to solve the least squares problem. This can potentially happen in the case of widely scattered data, where some of the knot panels do not contain data points. This problem can be solved, though, by using the matrix singular value decomposition.

Assume that a tensor product spline has been computed as described in Sec. 4.2.2. A least squares fit can be done to the scattered data as follows. Since the matrix A could be rank deficient, with potentially zero rows or columns, we cannot rely on the standard normal equations for determining the coefficients, but using the matrix singular value decomposition provides a convenient substitute (see Golub and Van Loan [15]). In the singular value decomposition of the matrix A , with the number of rows larger than the number of columns, the matrix is decomposed into the product of a column-orthogonal matrix U , a diagonal matrix S , with diagonal elements S_i , and the transpose V' of a square orthogonal matrix, so that $A = USV'$. In order to solve the problem $Ac = z$ we compute $c = V(S^+(U'z))$, where

$$S_i^+ = \begin{cases} \frac{1}{S_i} & S_i \neq 0 \\ 0 & \text{otherwise} \end{cases} \quad (23)$$

defines the generalized inverse of S and U' is the transpose of U . A tolerance is used to determine which of the small singular values in the decomposition should be considered zero.

4.3 A Knot Selection Algorithm

In a least squares data fitting process involving B-spline basis functions, the resulting model residuals are sensitive to the knot placement for the B-splines. The selection of B-spline knots in order to achieve as small a residual as possible during the least squares process is a nontrivial task. One would be extremely lucky to manually select a set of knots that could achieve a very small least squares residual. In this section we will describe a heuristic algorithm that, in practice, generates a set of knots that produces small

least squares residuals. The strategy involves an iterative knot insertion algorithm. An initial set of knots is selected by the algorithm user and, at each iteration of the algorithm, new knots are inserted in the vicinity of the previous fit where the local residuals are the largest. The knots are not moved once they have been inserted. The iterative algorithm has a stopping criteria based on a statistical test.

First we will discuss the knot insertion algorithm. It is based on a strategy suggested by Dierckx [3]. At the beginning of each iteration the assumption is that there exists a current set of knots. In the first iteration of the algorithm these would be an initial set chosen by the user. The tensor product of the B-splines is computed at the data points, a least squares fit is made to the data, and the current absolute residuals of the fit are computed at each data point. The current knots divide the (x, y) plane into rectangles. Some rectangles have data points and others don't. Let the knots at the k -th iteration be labeled $a < \xi_1^{(k)} < \xi_2^{(k)} < \dots < \xi_{r_k}^{(k)} < b$ along the x -axis and $c < \eta_1^{(k)} < \eta_2^{(k)} < \dots < \eta_{s_k}^{(k)} < d$ along the y -axis. The $(k+1)$ -iteration begins by associating all of the data points with the knot panels in which they fall. Let the index pair ij indicate the R_{ij} -th panel defined by $\xi_i \leq x \leq \xi_{i+1}$, $\eta_j \leq y \leq \eta_{j+1}$. Then suppose the data values $(x_1^{(ij)}, y_1^{(ij)}), \dots, (x_{n_{ij}}^{(ij)}, y_{n_{ij}}^{(ij)})$, fall into the R_{ij} -th panel. Let $F_k(x, y)$ be the least squares B-spline function fit to the data at the k -th iteration. We next compute the residuals of the fit at all of the data points

$$\text{Resid}_{kq} = z_q - F_k(x_q, y_q), \quad q = 1, \dots, n. \quad (24)$$

From these residuals we form the sums of squares of the residuals that fall within the knot intervals. The sums are separated into the x direction and the y direction as follows.

$$\begin{aligned} \delta_{i,x}^{(k)} &= \sum_q \left\{ \text{Resid}_{kq}^2 : \xi_i^{(k)} \leq x_q < \xi_{i+1}^{(k)} \right\}, \quad i = 1, 2, \dots, r_k, \\ \delta_{j,y}^{(k)} &= \sum_q \left\{ \text{Resid}_{kq}^2 : \eta_j^{(k)} \leq y_q < \eta_{j+1}^{(k)} \right\}, \quad j = 1, 2, \dots, s_k. \end{aligned} \quad (25)$$

What is meant here, for example in the case of $\delta_{i,x}^{(k)}$, is that the sum, at the k -th iteration of Resid_{kq}^2 , is taken over all pairs of points (x_q, y_q) such that $\xi_i^{(k)} \leq x_q < \xi_{i+1}^{(k)}$. This is done for each $i = 1, 2, \dots, r_k$.

We next find the maximums of these sums of squared residuals.

$$\begin{aligned} \delta_{u,x}^{(k)} &= \max \left\{ \delta_{i,x}^{(k)} : i = 1, 2, \dots, r_k \right\}, \\ \delta_{v,y}^{(k)} &= \max \left\{ \delta_{j,y}^{(k)} : j = 1, 2, \dots, s_k \right\}, \end{aligned} \quad (26)$$

where $u = i$ for some $i = 1, 2, \dots, r_k$ and $v = j$ for some $j = 1, 2, \dots, s_k$. The next step is to add one knot at a time at each iteration. In particular, a knot is added in the x direction, if $\delta_{u,x}^{(k)} > \delta_{v,y}^{(k)}$ or added in the y direction,

if $\delta_{u,x}^{(k)} < \delta_{v,y}^{(k)}$. The knot added is positioned based upon a weighted average of x or y data points in the columns or rows determined by the knot intervals with the maximum residual errors determined by (26). In particular, the positions are given by

$$\begin{aligned} \xi^{(k+1)} &= \sum_q \left\{ \frac{\text{Resid}_{kq}^2}{\delta_{u,x}^{(k)}} x_q : \xi_u^{(k)} \leq x_q < \xi_{u+1}^{(k)} \right\}, \\ \eta^{(k+1)} &= \sum_q \left\{ \frac{\text{Resid}_{kq}^2}{\delta_{v,y}^{(k)}} y_q : \eta_v^{(k)} \leq y_q < \eta_{v+1}^{(k)} \right\}. \end{aligned} \quad (27)$$

We note that

$$\begin{aligned} \sum_q \left\{ \frac{\text{Resid}_{kq}^2}{\delta_{u,x}^{(k)}} x : \xi_u^{(k)} \leq x_q < \xi_{u+1}^{(k)} \right\} &= 1, \\ \sum_q \left\{ \frac{\text{Resid}_{kq}^2}{\delta_{v,y}^{(k)}} : \eta_v^{(k)} \leq y_q < \eta_{v+1}^{(k)} \right\} &= 1, \end{aligned} \quad (28)$$

so that (27) represents weighted averages of all of the data pairs (x_q, y_q) satisfying $\xi_u^{(k)} \leq x_q < \xi_{u+1}^{(k)}$ and $\eta_v^{(k)} \leq y_q < \eta_{v+1}^{(k)}$.

The knots are then reindexed as necessary and $F_{k+1}(x, y)$ is computed as the least squares B-spline function fit to the data at the $(k+1)$ -iteration based on the new knot set. The iterations continue until the stopping criteria is met.

The stopping criteria for the k -th iteration used in this algorithm is based on the use of the R^2 -statistic, called the coefficient of multiple determination (see Draper and Smith [8]). R^2 is the square of the correlation

between the vector of observed data, z_q , $q = 1, \dots, n$, and the least squares predicted data, $F_k(x_q, y_q)$, $q = 1, \dots, n$. The statistic satisfies $0 \leq R^2 \leq 1$. This statistic is often used as a measure of how well the regression equation explains the variation in the data. It is known that in building models based on adding terms in the regression equation, care must be taken in using this statistic. However, in the current algorithm, the statistic is used in a somewhat non-conventional manner. We use it as a measure of the benefit of adding more knots to the tensor product spline model. The knot selection algorithm is terminated when $R^2 > 0.98$.

Although the knot placement algorithm is heuristic, coupling it with the R^2 -statistic has shown good convergence in practice. It is reasonable to expect this, since knots are placed in intervals in which the fits at a previous iteration showed the largest local error. Placing a knot there allows extra flexibility for those areas.

4.4 Data Smoothing by Tikhonov Regularization

As noted in Sec. 4.2.3, the data distribution can lead to a rank deficient least squares matrix. Even though a fit can be computed using SVD, evaluations of the fitted function at some points can lead to unwanted oscillations. It is necessary to introduce a smoothing procedure by regularization. Regularization is a way of introducing extra information to the least squares objective function in order to control overfitting of the data. It is the overfitting of data that can lead to the unwanted oscillations. In the present work a penalty term is introduced to control the smoothness of the resulting fitted function. The objective function will then balance the data fitting with the smoothness of the fit. The second partial derivatives of the tensor product B-splines will be introduced as the smoothing operators. The objective function will take the form

$$O(c) = \sum_{q=1}^n \left[\sum_{i=1}^r \sum_{j=1}^s c_{ij} N_i(x_q) N_j(y_q) - z_q \right]^2 + \sum_{p=1}^t \left[\sum_{i=1}^r \sum_{j=1}^s \lambda c_{ij} N_i''(u_p) N_j(v_p) \right]^2 + \sum_{p=1}^t \left[\sum_{i=1}^r \sum_{j=1}^s \lambda c_{ij} N_i(u_p) N_j''(v_p) \right]^2. \quad (29)$$

λ is called a Tikhonov parameter.

In the objective function the data values are given by (x_q, y_q, z_q) , $q = 1, \dots, n$. The smoothing terms will be

evaluated at a new set of points chosen so that every knot panel has an equal number of points assigned to the panel. In the current case there will be a total of t points throughout the knot panels given by (u_p, v_p) , $p = 1, \dots, t$.

To write this in matrix form we will define two new matrices B_1 and B_2 as follows. For $m = 1, 2, \dots, rs$ let (i_m, j_m) be such that $m = (j_m - 1)r + i_m$. Then define the matrix elements

$$\begin{aligned} A(q, m) &= N_{i_m}(x_q) N_{j_m}(y_q) \\ B_1(p, m) &= N_{i_m}''(u_p) N_{j_m}(v_p) \\ B_2(p, m) &= N_{i_m}(u_p) N_{j_m}''(v_p). \end{aligned} \quad (30)$$

We can rewrite (29) in the form

$$\begin{aligned} O(c) &= \sum_{q=1}^n \left[\sum_{m=1}^{rs} c_m A(q, m) - z_q \right]^2 \\ &\quad + \sum_{p=1}^t \left[\lambda \sum_{m=1}^{rs} c_m B_1(p, m) \right]^2 \\ &\quad + \sum_{p=1}^t \left[\lambda \sum_{m=1}^{rs} c_m B_2(p, m) \right]^2 \\ &= \|Ac - z\|^2 + \|\lambda B_1 c\|^2 + \|\lambda B_2 c\|^2 \\ &= \left\| \begin{pmatrix} A \\ \lambda B_1 \\ \lambda B_2 \end{pmatrix} c - \begin{pmatrix} z \\ 0 \\ 0 \end{pmatrix} \right\|^2. \end{aligned} \quad (31)$$

The matrix dimensions in this objective function are:

$$A \text{ is } n \times rs, B_1 \text{ is } t \times rs, B_2 \text{ is } t \times rs, \begin{pmatrix} A \\ \lambda B_1 \\ \lambda B_2 \end{pmatrix} \text{ is } (n+2t) \times rs, \text{ and } \begin{pmatrix} z \\ 0 \\ 0 \end{pmatrix} \text{ is } (n+2t) \times 1.$$

If we perform a QR decomposition, then

$$\begin{pmatrix} A \\ \lambda B_1 \\ \lambda B_2 \end{pmatrix} = QR, \quad (32)$$

where Q is $(n+2t) \times (n+2t)$ and R is $(n+2t) \times rs$. Q is orthogonal, so that $Q^T Q = I$, and R is zero except in its

upper right corner. Let \hat{R} be the $rs \times rs$ upper triangular portion of R and let

$$d = Q^T \begin{pmatrix} z \\ 0 \\ 0 \end{pmatrix}. \quad (33)$$

Then, let \hat{d} be the upper rs entries of d . The minimum of $O(c)$ then satisfies $\hat{R}c = \hat{d}$. For a more complete discussion of the QR method see Golub and Van Loan [15].

At this point we need to discuss the selection of the Tikhonov parameter, λ . A number of methods for choosing the parameter have been discussed in the literature (see [16, 17, 18, 19, 20]). For this work, though, we used a graphical method that led to the selection of a parameter in a few iterations. Before continuing to discuss the selection of the Tikhonov parameter for the current study we need to change the coordinates of the original probe points to spherical coordinates defined on a rectangle.

4.4.1 A Coordinate Transformation

Since our tensor product B-spline requires a rectangular grid, we convert our CMM data to spherical coordinates. We start with a given set of n points, (x_i, y_i, z_i) , $i = 1, 2, \dots, n$, for some n , on a surface. As measured, these points are given relative to the CMM origin. The first step is to center the data by using the center-of-data mass of the data points. This establishes an origin at the center-of-data mass point. It is done in order to establish a common reference point interior to the measured data points. It also simplifies writing vectors from the origin to the data points and allows the introduction of spherical coordinates. The center-of-data point is given by

$$\begin{aligned} \bar{x} &= \frac{1}{n} \sum_{i=1}^n x_i \\ \bar{y} &= \frac{1}{n} \sum_{i=1}^n y_i \\ \bar{z} &= \frac{1}{n} \sum_{i=1}^n z_i. \end{aligned} \quad (34)$$

Since the data set is enclosed in a near-spherical bounded region, it is reasonable to identify the Euclidean data points with spherical coordinates. This will map the points on the sphere to a rectangular surface where the surface coordinates are designated by θ, ϕ and the height of a surface point is given by $R(\phi, \theta)$. In order to use spherical coordinates to represent points on the boundary of a surface, we need to restrict our analysis to surfaces that are called star-shaped. These are surfaces in which a ray drawn from the center-of-data mass intersects the boundary in a unique point. Whereas, in the definition of the B-splines, we used the coordinates (x, y) we will now use (ϕ, θ) and build B-splines in terms of these spherical coordinates. Euclidean coordinates will now refer to the measured data points. This switch in notation should not, it is hoped, cause too much confusion.

To each data point there is a vector from $(\bar{x}, \bar{y}, \bar{z})$ to (x_i, y_i, z_i) , given by $V_i = (x_i, y_i, z_i) - (\bar{x}, \bar{y}, \bar{z})$. Furthermore, any point within the bounding sphere can be identified by spherical coordinates of the form $S(R, \phi, \theta) = (R \sin(\theta) \cos(\phi), R \sin(\theta) \sin(\phi), R \cos(\theta))$, where $x = R \sin(\theta) \cos(\phi)$, $y = R \sin(\theta) \sin(\phi)$, $z = R \cos(\theta)$, for $0 \leq \theta \leq \pi$, $0 \leq \phi \leq 2\pi$. θ is referred to as the colatitude and ϕ is referred to as the azimuth (Fig. 8). Table 2 associates the three dimensional octants with their spherical coordinates (where we have suppressed R). Therefore, the Euclidean coordinates were converted to θ, ϕ angles on a rectangle $[0, \pi] \times [0, 2\pi]$. The height, $R(\phi, \theta)$, at each θ, ϕ is taken as the estimated radius from the center-of-data mass of all of the Euclidean coordinates.

As an illustration, the results of the conversion of the coarse probe points for the Green phantom from Euclidean coordinates to ϕ, θ coordinates are shown in Fig. 9. We note the density of data points near the equator is higher than towards the poles at $\theta = 0$ and $\theta = \pi$. Unfortunately the distribution of data points was dictated by the software controlling the CMM. The lack of data points near the poles leads to a well known problem, called the *Pole Problem* in the literature (see Dierckx [3]). It creates rank deficient matrices during the least squares fitting process. Section 4.2.3 discussed how the singular value decomposition can be used to handle this problem.

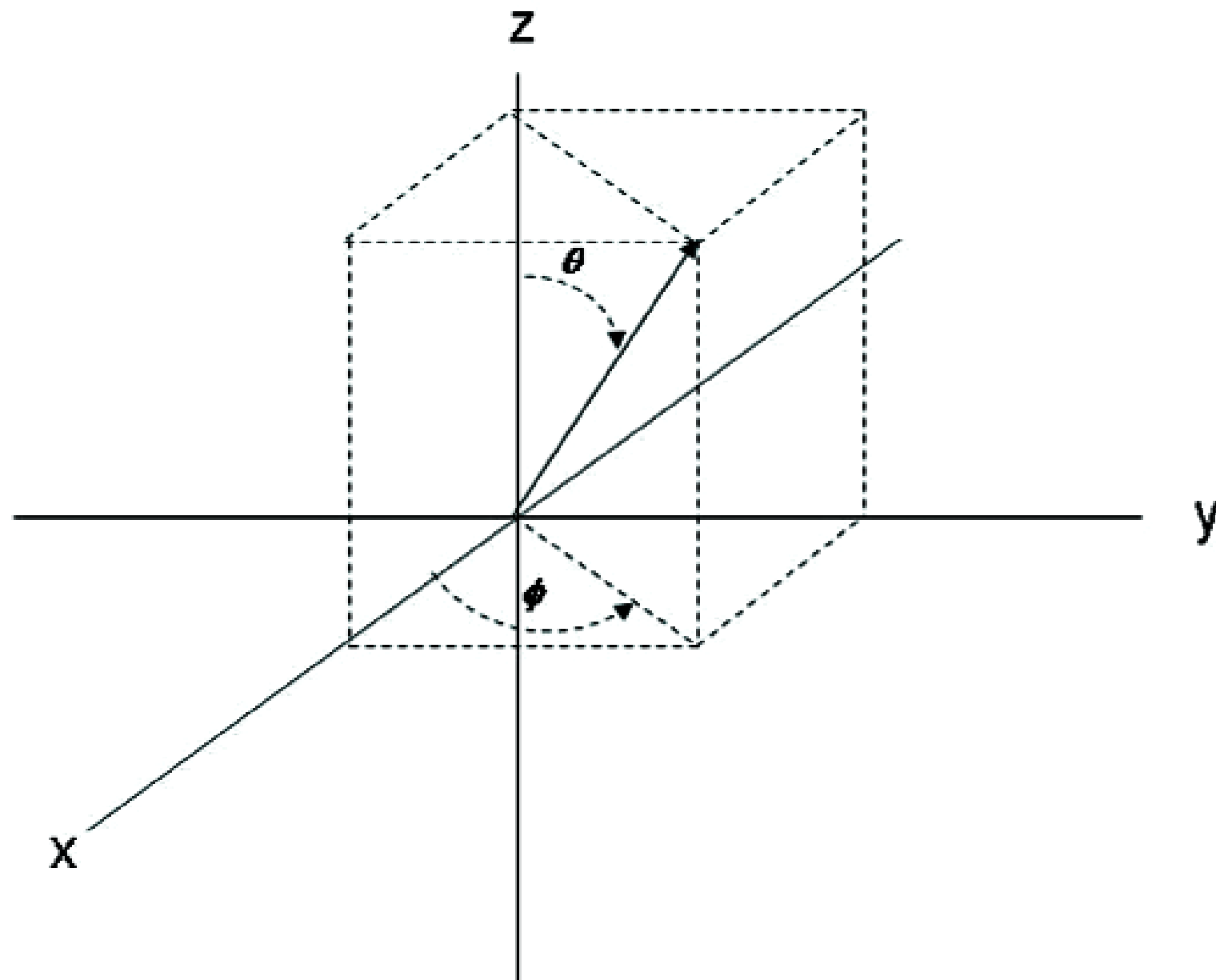


Fig. 8. Spherical Coordinate angles. θ is the colatitude and ϕ is the azimuth.

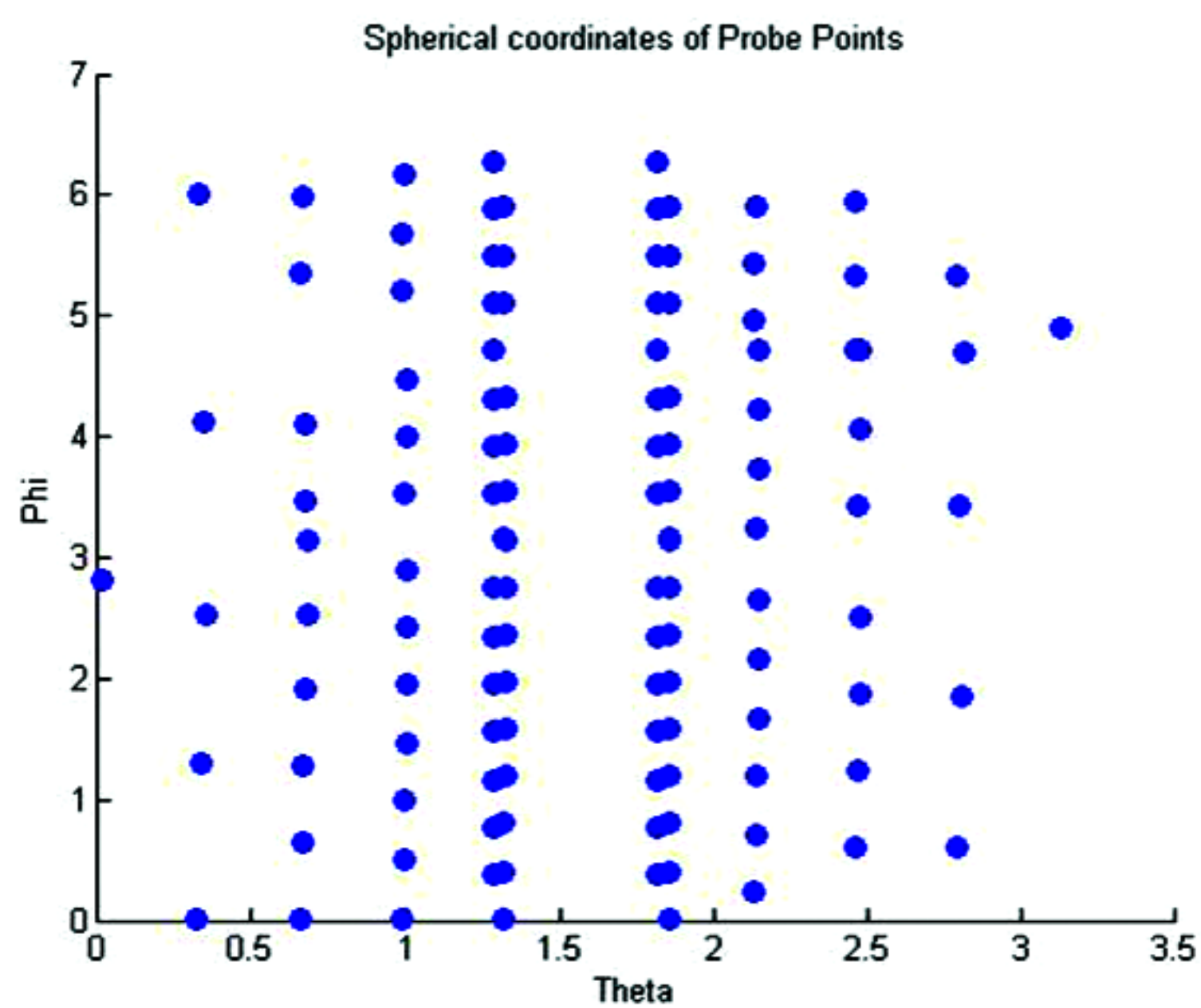


Fig. 9. Plot of the θ , ϕ Coordinates of the Probe Points for the Coarse Data.

Table 2. Octant Equivalence between Euclidean and Spherical Coordinates

Octant Number	3-D Octants to Spherical Coordinates	
	Cartesian Coordinate	Spherical Coordinate
1	$x \geq 0, y \geq 0, z \geq 0$	$0 \leq \theta \leq \pi/2, 0 \leq \phi \leq \pi/2$
2	$x < 0, y \geq 0, z \geq 0$	$0 \leq \theta \leq \pi/2, \pi/2 \leq \phi \leq \pi$
3	$x < 0, y < 0, z \geq 0$	$0 \leq \theta \leq \pi/2, \pi \leq \phi \leq 3\pi/2$
4	$x \geq 0, y < 0, z \geq 0$	$0 \leq \theta \leq \pi/2, 3\pi/2 \leq \phi \leq 2\pi$
5	$x \geq 0, y \geq 0, z < 0$	$\pi/2 \leq \theta \leq \pi, 0 \leq \phi \leq \pi/2$
6	$x < 0, y \geq 0, z < 0$	$\pi/2 \leq \theta \leq \pi, \pi/2 \leq \phi \leq \pi$
7	$x < 0, y < 0, z < 0$	$\pi/2 \leq \theta \leq \pi, \pi \leq \phi \leq 3\pi/2$
8	$x \geq 0, y < 0, z < 0$	$\pi/2 \leq \theta \leq \pi, 3\pi/2 \leq \phi \leq 2\pi$

4.4.2 Choosing a Tikhonov Parameter

Once the original probe data had been converted to spherical coordinates, the iterative selection of the Tikhonov parameter for the current volume estimation problem proceeded as follows. First, a set of knots was selected with $\lambda = 0$ (i.e., no regularization terms) in order to produce an $R^2 > 0.98$ as described in Sec. 4.3. We found that, for all of the data sets examined, only a very few extra knots were added beyond the initial set used to begin the knot selection process. This led to a rapid convergence to a set of final knots in all cases. These knots formed panels in the rectangle $[0, \pi] \times [0, 2\pi]$. Next, nine points were uniformly chosen in each panel and the regularization terms formed. Numerical experimentation showed that the use of nine points in each panel provided sufficient extra data to the regularization terms in order to smooth the final fits. An initial Tikhonov parameter, λ , was chosen and the objective function (29) was minimized by the *QR* method discussed above. An initial grid of

40 θ and 80 ϕ values was generated in the rectangle $[0, \pi] \times [0, 2\pi]$. This grid was finally fixed on for selecting a Tikhonov parameter since further numerical experimentation showed that the response surface of the radii for denser grids did not change the final range of the radii. The coefficients, c , computed to minimize the objective function (29), were then inserted into the linear form Ac , where A was the tensor product matrix formed from all of the 40×80 grid points. The end resulting radii at the grid values were then computed and the spherical volumes for those radii were computed. The maximum and minimum sphere volumes over the entire 40×80 grid were determined. To select the appropriate Tikhonov parameter, λ values over a range were used and the differences between the maximum and minimum volumes were plotted. The value of λ that produced the minimum difference was selected as the working λ . For the current volume estimation problem, the final λ was $\lambda = 0.32$.

To illustrate the effect of using regularization terms to smooth the least squares fit of the radii see Fig. 10.

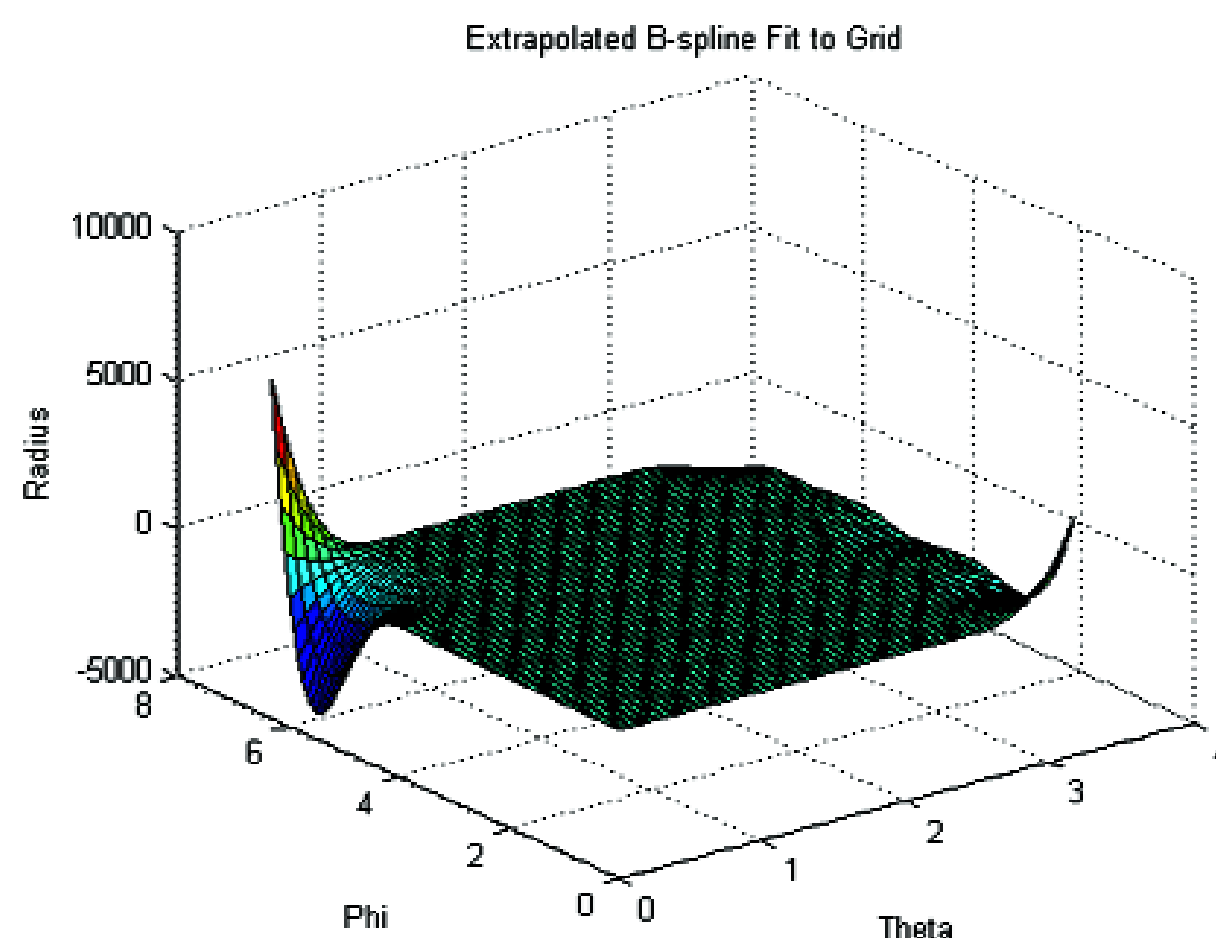


Fig. 10. Plot of Radius Data on a 40 by 80 Grid. $\lambda = 0$.

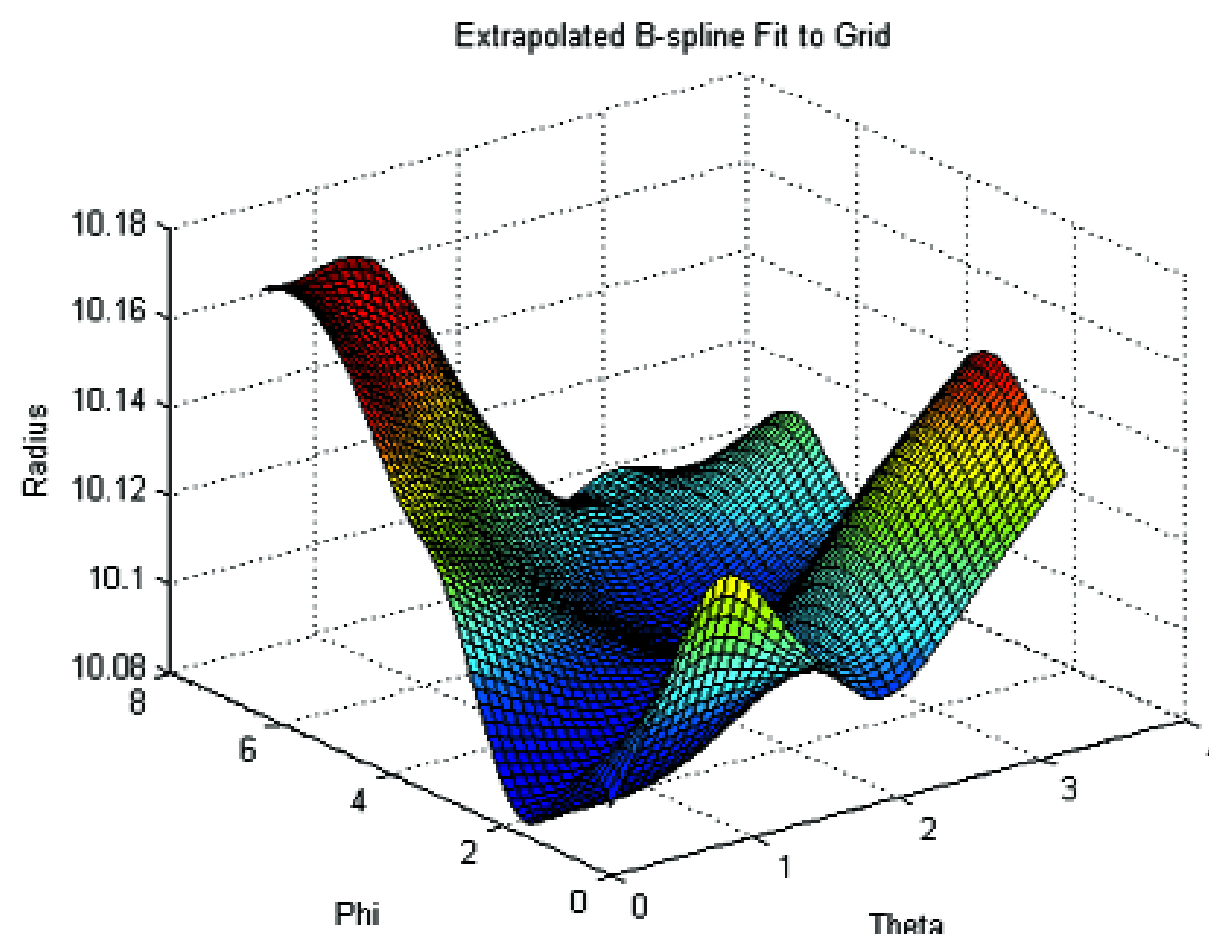


Fig. 11. Plot of Radius Data on a 40 by 80 Grid. $\lambda = 0.32$.

This figure shows the radii data for the case of $\lambda = 0$, or no regularization. Note the large radii oscillations at the boundaries (a range of approximately 10,000 mm). Now see Fig. 11 and note the narrow range (approximately 0.1 mm) of the radii over the entire grid. This clearly shows the effect of the regularization terms on the least squares fit. The data used for these plots was the coarse Green phantom data.

5. Volume Estimation by Surface Triangulation

We could now estimate the volume using the triangulation method described in Sec. 2 by dividing the phantom surfaces into triangular surface patches as follows. First of all we partitioned the phantom surfaces at grid points located at the colatitude angles $\theta_1 = 0 < \theta_2 < \dots < \theta_v < \theta_{v+1} = \pi$ from the north pole to the south pole and azimuthal angles $\phi_1 = 0 < \phi_2 < \dots < \phi_h < \phi_{h+1} = 2\pi$ around the phantom surface, where v stands for vertical and h for horizontal. Since these grid points did not necessarily fall at the measured data points, a radius, $R(\phi, \theta)$, was calculated at the grid points from the regularized fitted B-spline surface model. At the north and south poles the radius was taken as the median value, R_{north} , of the grid point values $R(\phi_i, 0)$, $i = 1, \dots, h$, for the north pole and R_{south} , the median value of $R(\phi_i, \pi)$, $i = 1, \dots, h$. The spherical coordinates of all of the grid points were converted to Euclidean coordinates on the surface by

$$\begin{aligned} x &= R \sin(\theta) \cos(\phi) \quad 0 \leq \theta \leq \pi, \\ y &= R \sin(\theta) \sin(\phi) \quad 0 \leq \phi \leq 2\pi, \\ z &= R \cos(\theta). \end{aligned} \quad (35)$$

At this point we could apply the Divergence Theorem method of Sec. 2. The patches at the north poles were easily constructed to be triangular as part of the process of determining the contribution of each patch to the phantom volumes. In particular, at the north pole the designated point was $(x_1, y_1, z_1) = (0, 0, R_{north})$. We then iterated through the spherical coordinate points (ϕ_i, θ_2) , $i = 1, \dots, h$. At each of these angle pairs there was a value $R(\phi_i, \theta_2)$, $i = 1, \dots, h$. We generated the volume by adding up the contributions of each patch to the volume total. We did this by initializing a variable, vol , for the volume, to zero. We then started to generate the contribution from the first layer of patches at the north pole. As noted above, we set the north pole to (x_1, y_1, z_1) and then set

$$\begin{aligned} x_2 &= R(\phi_1, \theta_2) \sin(\theta_2) \cos(\phi_1), \\ y_2 &= R(\phi_1, \theta_2) \sin(\theta_2) \sin(\phi_1), \\ z_2 &= R(\phi_1, \theta_2) \cos(\theta_2), \end{aligned}$$

and

$$\begin{aligned} x_3 &= R(\phi_2, \theta_2) \sin(\theta_2) \cos(\phi_2), \\ y_3 &= R(\phi_2, \theta_2) \sin(\theta_2) \sin(\phi_2), \\ z_3 &= R(\phi_2, \theta_2) \cos(\theta_2). \end{aligned}$$

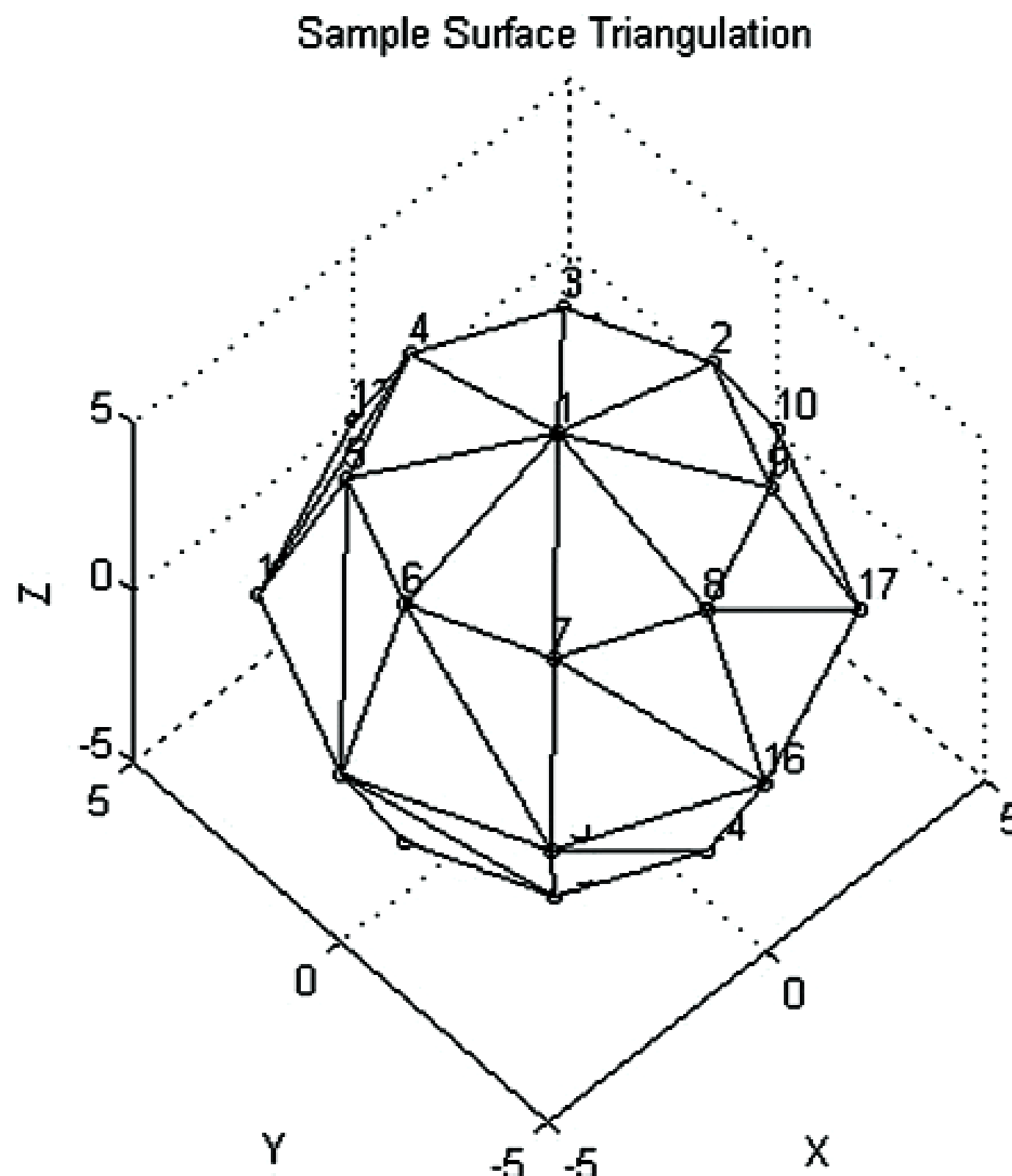


Fig. 12. A Sample Triangulation of a Sphere Viewed from the North Pole (Vertex 1).

These were set in a positive orientation.

In order to make this process more concrete, we have included a sample surface triangulation with $v=5$ colatitude angles and $h=8$ azimuthal angles in Fig. 12. In this figure the Euclidean grid points have been indexed. The North Pole (x_1, y_1, z_1) is designated by the index 1. In the first step described above the points (x_2, y_2, z_2) and (x_3, y_3, z_3) are indexed in Fig. 12 by points 2 and 3 respectively. Next we computed the outward normal to the triangle patch by forming the vectors $v_1 = (x_2, y_2, z_2) - (x_1, y_1, z_1)$, $v_2 = (x_3, y_3, z_3) - (x_1, y_1, z_1)$, for triangle 123 in Fig. 12 and then formed the normalized cross product

$$\hat{n} = \frac{v_1 \times v_2}{\|v_1 \times v_2\|}. \quad (36)$$

We then computed the contribution that this patch made to the volume as

$$\text{vol} = \text{vol} + (\hat{n} \cdot (x_1, y_1, z_1))$$

$$\left\{ \hat{n} \cdot \left(\sum_{j=1}^3 (y_j z_{j+1-k} - y_{j+1-k} z_j), (z_j x_{j+1-k} - z_{j+1-k} x_j), (x_j y_{j+1-k} - x_{j+1-k} y_j) \right) \right\}, \quad (37)$$

where

$$k = \begin{cases} 0 & 1 \leq j < 3 \\ 3 & j = 3 \end{cases} \quad (38)$$

is set in order to compensate for the cyclical vertex indexing around the triangle patch. Formula (37) is a combination of Eq. (6), where $(\hat{n} \cdot (x_1, y_1, z_1))$ is the coefficient c in (6), and the second factor is a compact form of Eq. (11). The factors $1/3$ from (6) and $1/2$ from (11) were combined as a multiple of $1/6$ after all of the summations had been performed.

We proceeded to the next patch in the North Pole layer. Again (x_1, y_1, z_1) was the North Pole, indexed by 1 in Fig. 12, and we then used the previous computation to get $x_2 = x_3, y_2 = y_3, z_2 = z_3$ and set

$$\begin{aligned} x_3 &= R(\phi_3, \theta_2) \sin(\theta_2) \cos(\phi_3), \\ y_3 &= R(\phi_3, \theta_2) \sin(\theta_2) \sin(\phi_3), \\ z_3 &= R(\phi_3, \theta_2) \cos(\theta_2). \end{aligned}$$

In Fig. 12 the new point (x_3, y_3, z_3) is indexed by 4 in Fig. 12. The triangle of interest is now 134 in terms of indices. We computed the normalized cross product as for the first patch and then computed the contribution of the second patch to the volume using Eq. (37). We continued this process for $\theta_2, \phi_i, i = 1, \dots, h$. In Fig. 12 we would have proceeded with computing contributions to the volume by working through the indexed triangles 123, 134, 145, 156, 167, 178, 189, and 192.

We next computed the contributions of the middle layer patches in a two step process. We iterated through each $\theta_j, j = 2, \dots, v-1$. The triangles at the South Pole were handled separately. For each $\theta_j, j = 2, \dots, v-1$ and $\phi_i, i = 1, \dots, h$, the patches were defined first in terms of four vertices to create four sided patches. Each of these patches was then divided into two triangles. The four vertices of a rectangular patch were identified counterclockwise as $(\theta_j, \phi_i, R(i, j)), (\theta_{j+1}, \phi_i, R(i, j+1)), (\theta_{j+1}, \phi_{i+1}, R(i+1, j+1)), (\theta_j, \phi_{i+1}, R(i+1, j))$. As an example, in Fig. 12 one of the patches is identified by indices, in counterclockwise order, as 8 16 17 9. These indices would be associated with vertices $(\theta_2, \phi_1, R(1, 2)), (\theta_3, \phi_1, R(1, 3)), (\theta_3, \phi_2, R(2, 3)), (\theta_2, \phi_2, R(2, 2))$. The four vertex patches were then divided into two triangles. For the first triangle in the rectangular patch we set

$$\begin{aligned} x_1 &= R(i, j) \sin(\theta_j) \cos(\phi_i), \\ y_1 &= R(i, j) \sin(\theta_j) \sin(\phi_i), \\ z_1 &= R(i, j) \cos(\theta_j). \end{aligned}$$

$$\begin{aligned} x_2 &= R(i, j+1) \sin(\theta_{j+1}) \cos(\phi_i), \\ y_2 &= R(i, j+1) \sin(\theta_{j+1}) \sin(\phi_i), \\ z_2 &= R(i, j+1) \cos(\theta_{j+1}). \end{aligned}$$

$$\begin{aligned} x_3 &= R(i+1, j+1) \sin(\theta_{j+1}) \cos(\phi_{i+1}), \\ y_3 &= R(i+1, j+1) \sin(\theta_{j+1}) \sin(\phi_{i+1}), \\ z_3 &= R(i+1, j+1) \cos(\theta_{j+1}). \end{aligned}$$

This triangle in the example Fig. 12 would be 8 16 17. Again, the volume increment was computed as discussed previously. For the second triangle of the rectangular patch we maintained the same (x_1, y_1, z_1) and set $x_2 = x_3, y_2 = y_3, z_2 = z_3$ and then set

$$\begin{aligned} x_3 &= R(i+1, j) \sin(\theta_j) \cos(\phi_{i+1}), \\ y_3 &= R(i+1, j) \sin(\theta_j) \sin(\phi_{i+1}), \\ z_3 &= R(i+1, j) \cos(\theta_j). \end{aligned}$$

This triangle in the example Fig. 12 would be 8 17 9. Again the volume increment was computed as before. We continued the process for $\theta_j, j = 2, \dots, v-1$ and $\phi_i, i = 1, \dots, h$.

Finally, at the South Pole there were h triangles to include in the volume calculation. Their vertices were identified as follows

$$\begin{aligned} x_1 &= R(i, v) \sin(\theta_v) \cos(\phi_i), \\ y_1 &= R(i, v) \sin(\theta_v) \sin(\phi_i), \\ z_1 &= R(i, v) \cos(\theta_v). \end{aligned}$$

$$\begin{aligned} x_2 &= 0, \\ y_2 &= 0, \\ z_2 &= -R_{\text{south}}. \end{aligned}$$

$$\begin{aligned} x_3 &= R(i+1, v) \sin(\theta_v) \cos(\phi_{i+1}), \\ y_3 &= R(i+1, v) \sin(\theta_v) \sin(\phi_{i+1}), \\ z_3 &= R(i+1, v) \cos(\theta_v). \end{aligned}$$

As an example, in Fig. 13 one of these triangles would be indexed by the points 23 26 24, where the South Pole is point number 26, although the South Pole index may be hard to see in the figure. The contribution of these triangles to the volume was computed as above. After all of the triangle contributions to the volume were computed, the final volume was taken to be $\text{vol} = (1/6) \text{vol}$. The factor 1/6 comes from the product of 1/3 from Sec. 2.1 and 1/2 from Sec. 2.2. The reader can refer back to these sections to see how the factors arose. The triangulation process can clearly be generalized to a denser surface triangulation.

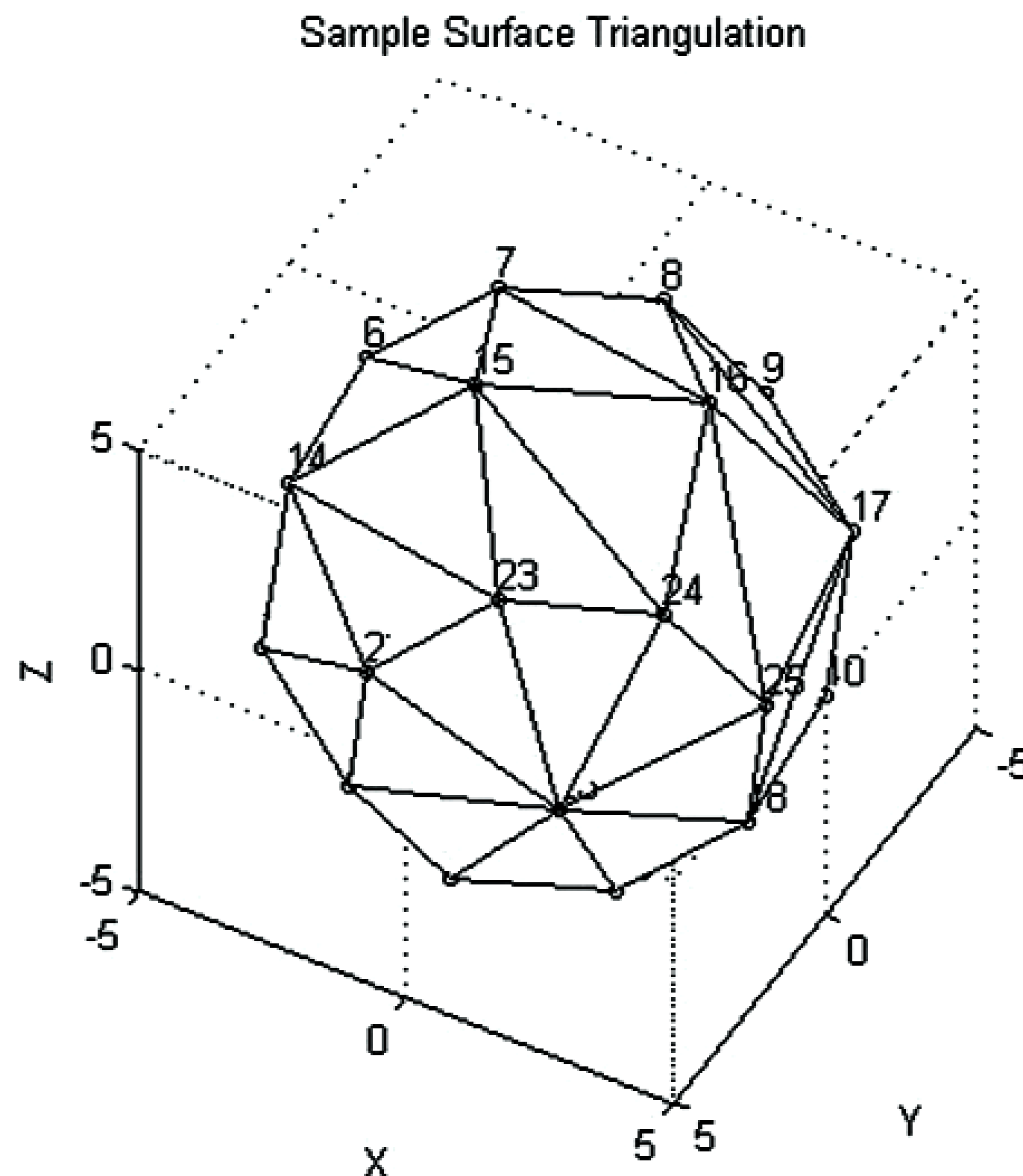


Fig. 13. A Sample Triangulation of a Sphere Viewed from the South Pole (Vertex 26).

6. Computational Results

Two computational processes were involved in generating the results for each object. The objects involved were the calibrated sphere and the two phantoms. First, sphere models were fit to an object. The second process involved three steps. The first step was to compute a good selection of knots for the tensor product spline function. The next step was to fit the tensor product spline function with the objective function modified by the regularization terms to the object. Finally, once the tensor spline function had been developed they were used to generate data at the vertices of the surface triangulation and to compute the volume using the Divergence Theorem.

6.1 Computational Results for the Spherical Model

Tables 3, 4, and 5 present the results obtained by fitting spherical models to the data from the calibrated sphere and the two phantoms. For the calibrated sphere there were five repeats for each of the coarse and dense

data sets. Since the results of the point location repeats differed only at the micrometer level the data sets were combined by averaging to form two data sets representing the coarse and dense data sets for the calibrated sphere. Similarly, the point locations of the three position data sets for both the coarse and dense data sets for the phantoms differed only at the micrometer level. Therefore, by averaging of the three position data sets for coarse and dense data sets, two working data sets were formed for the Green and Pink phantoms. Since the output of the Nelder-Mead algorithm was a final polytope surrounding the minimizing parameter with the polytope, the final reported parameters were selected as the median value of the vertex values. These are the first four entries in the tables: Center x, Center y, Center z, and Radius. The units are millimeters. The fifth table entries are the spherical volumes based on the median radius value in cubic millimeters. The radius residuals were computed as the absolute value of

$$\text{Resid}_i = \sqrt{(x_i - \hat{c}_1)^2 + (y_i - \hat{c}_2)^2 + (z_i - \hat{c}_3)^2} - \hat{c}_4. \quad (39)$$

The sixth and seventh entries in the tables give the mean value and standard deviation of these residuals in millimeters. The eighth through the eleventh entries are the expanded uncertainties for the estimated Center x, Center y, Center z, and Radius, also in millimeters.

The results in these sections on spherical model fitting involved the non-linear model (39) fitting to the sparse data sets generated by the CMM. This process did not involve the B-spline algorithm. Therefore any differences between coarse and dense results were most likely the consequence of the point selection in the metrology of the artifacts.

6.1.1 Results for the Spherical Model Fit to the Calibrated Sphere With Coarse and Dense Data

Table 3 gives the results of the fit of the sphere model to both the coarse (122 points) and the dense (181 points) data sets for the calibrated sphere. In both cases the average of the five repeat data sets was used for the fitting process. As can be seen, the two volume estimates differ by approximately 0.04 %. The radii estimates differ by about 0.01 %. There is a slightly larger difference when the spherical fit results are compared to the volume estimate based on the manufacturer estimated calibrated sphere diameter of 19.05 mm. This would lead to a volume estimate of 3619.8 mm³. Since the method of estimating the diameter by the manufacturer was proprietary we could not independently verify the measurement. We could only compare it against our sphere model fit results. The manufacturer measured radius differs from the computed radii in Table 3 by about 0.08 % whereas the volume estimate based on the measured radius differs from the volume estimates in Table 3 by about 0.2 %. If we expand the radii in Table 3 to a diameter we find the values 19.0651 millimeters in the coarse case and 19.0629 millimeters in the dense case. These differ by approximately 0.01 mm from the manufacture's measured diameter. It would seem that these differences were sufficiently small so that the difference in value estimates for radius and volume were not considered significant. Although the estimated centers are slightly different, all other values are of the same order of magnitude. Based on these results we can accept that the CMM is producing accurate position data for spherical metallic artifacts. Tables 4 and 5, however, begin to show the consequences involved with attempting to measure slightly non-spherical and non-metallic artifacts with the CMM.

6.1.2 Results for the Spherical Model Fit to the Phantoms With Coarse and Dense Data Sets

From Tables 4 and 5 it is clear that the Green phantom is more spherical than the Pink phantom as indicated by the residuals and the expanded uncertainties. This simply confirms the fact that the Pink phantom was more difficult to measure using the vacuum chuck due to its lack of sphericity. For example, the Mean Radial Residual for the Pink data is two orders of magnitude larger than for the Green data. The Standard Deviations for the Pink data are an order of magnitude greater, as are the Expanded Uncertainties. It is not clear how much the phantom material affected the results since it was difficult to set up a specific probe test for metallic versus phantom material. The artifacts would have to have been exactly the same size, positioned at exactly the right location, and probed at exactly the same coordinates to separate those factors from the material factor difference. There were no such comparable artifacts. We can make a guess, though, that there might be some effect due to probe force against the non-metallic material if we look at Table 3 and Table 4 for the Green phantom, the most spherical of the two phantoms. The Expanded Uncertainties for the sphere fit to the calibrated metallic sphere and the Expanded Uncertainties for the sphere fit to the Green phantom data differ by one to three orders of magnitude. Since the repeatability of the CMM measurements is at the 1 μ m level, it is likely then that material difference had some significant affect on the difference in the uncertainties. It is a conjecture that this and the non-spherical shape of the Pink phantom account for a large part of the differences between the Pink phantom Expanded Uncertainties in Table 5, the Green phantom Expanded Uncertainties in Table 4, and the calibrated sphere Expanded Uncertainties in Table 3. For the Green phantom the radii estimates in Table 4 between the fits of the coarse and dense data sets is approximately 0.3 % and the volumes differ by approximately 0.8 %. Whereas, for the Pink phantom the radii estimates in Table 5 between the fits of the coarse and dense data sets are 0.4 % and the volumes differ by approximately 1.3 %. In both Tables 4 and 5 the expanded uncertainties for the dense data sets are approximately an order of magnitude better than the results for the coarse data sets. This suggests that the radius and volume estimates in the case of the dense data sets for both the Green and Pink phantoms are probably the more realistic values, at least in terms of a spherical model fit.

Table 3. Results of a Spherical Fit to Coarse and Dense Data for Calibrated Sphere

Properties	Spherical Fit Results for Calibrated Sphere	
	Coarse	Dense
Center x	0.2803×10^{-4}	-0.1331×10^{-2}
Center y	-0.4881×10^{-3}	-0.2034×10^{-2}
Center z	-0.6642×10^{-2}	-0.2564×10^{-2}
Radius	9.5326	9.5314
Est. Volume	3628.41	3627.14
Mean Rad. Residual	0.3231×10^{-4}	0.8720×10^{-4}
Stand. Dev. Rad. Residual	0.2156×10^{-2}	0.22302×10^{-2}
Expanded Uncert. Center x	0.2711×10^{-4}	0.2357×10^{-4}
Expanded Uncert. Center y	0.2711×10^{-4}	0.2357×10^{-4}
Expanded Uncert. Center z	0.4316×10^{-4}	0.3757×10^{-4}
Expanded Uncert. Radius	0.2188×10^{-4}	0.1909×10^{-4}

Table 4. Results of a Spherical Fit to Coarse Data for the Green Phantom

Properties	Spherical Fit Results to Green Phantom	
	Coarse	Dense
Center x	0.8216×10^{-3}	-0.9172×10^{-3}
Center y	0.1532×10^{-2}	0.4275×10^{-2}
Center z	-0.1832×10^{-1}	0.1723
Radius	10.1121	10.0850
Est. Volume	4331.3	4296.48
Mean Rad. Residual	-1.4802×10^{-4}	-0.8450×10^{-4}
Stand. Dev. Rad. Residual	0.5526×10^{-1}	0.3093×10^{-1}
Expanded Uncert. Center x	0.1926×10^{-1}	0.4757×10^{-2}
Expanded Uncert. Center y	0.1926×10^{-1}	0.4753×10^{-2}
Expanded Uncert. Center z	0.2123×10^{-1}	0.7261×10^{-2}
Expanded Uncert. Radius	0.1146×10^{-1}	0.3632×10^{-2}

Table 5. Results of a Spherical Fit to Coarse Data for the Pink Phantom

Properties	Spherical Fit Results to Pink Phantom	
	Coarse	Dense
Center x	0.7110×10^{-3}	0.2495×10^{-1}
Center y	-0.1329×10^{-2}	0.3272×10^{-1}
Center z	0.1739×10^{-3}	0.3719
Radius	9.7332	9.7752
Est. Volume	3862.09	3912.54
Mean Rad. Residual	-0.2315×10^{-2}	-0.3223×10^{-3}
Stand. Dev. Rad. Residual	0.2127	0.8935×10^{-1}
Expanded Uncert. Center x	0.2686	0.3818×10^{-1}
Expanded Uncert. Center y	0.2693	0.3862×10^{-1}
Expanded Uncert. Center z	0.2972	0.5832×10^{-1}
Expanded Uncert. Radius	0.1602	0.2880×10^{-1}

Table 1 shows that the diameter of the Green phantom is approximately 20 mm and that the diameter of the Pink phantom falls approximately between 20.0 mm and 18.4 mm. This would imply that the sphere model for the Green phantom should have a volume of about 4188.8 mm^3 . With the estimate of 0.3 mm uncertainty using the calipers, this would place the Green volume in the range of 4380.1 mm^3 and 4003.1 mm^3 . The estimated volumes in Table 4 are within this range for both the coarse and the dense data. The Pink phantom should have a volume between 4445.2 mm^3 and 3104.8 mm^3 . Table 5 for the Pink phantom shows that for the coarse and dense data the volume estimates fall from about 3862 mm^3 to about 3913 mm^3 . These volumes are within the expected range of the caliper measurements.

6.2 Computational Results for the B-Spline Model

In this section we will describe the method used to estimate volume uncertainties and discuss the results of using a B-spline surface model and the Divergence Theorem to estimate the phantom volumes. A nonparametric method to estimate the volume uncertainties was chosen since there did not exist any “ground truth” values for the Green and Pink FDA phantom volumes. All of the expanded uncertainty results are displayed in row two of Tables 6, 7, and 8.

For each data set we conducted twenty one estimates of the volumes and volume uncertainties by increasing the density of the surface triangulation. The twenty one were selected since beyond that point the computer time became large and the volume increment per case was minimal. In order to plot the volume results in terms of grid density we use the term grid size by which we mean the product of the number of θ values times the number of ϕ values for a particular grid density. As shown in Tables 6 through 8 we begin with a grid of ten θ and twenty ϕ values and compute the volume and uncertainty. The grid size in this 10×20 grid is then 200. The θ values were then incremented by ten and the ϕ values by twenty for each grid case until the last case of 210 θ values and 420 ϕ values, giving a grid size of 88200 (8.82×10^4). The volumes in Figs. 14 through 19 grow very rapidly and appear to approach a fixed value as the grid size increases. They simply reflect the volume data in the Tables 6 through 8.

6.2.1 Estimating Volume Uncertainties

The uncertainties were estimated by the nonparametric “bootstrap” method. It is a computer intensive technique for estimating uncertainties and it involves repeated Monte Carlo resampling from the spherical coordinates of the original measured data sets with radii values modified by the fitting residuals, refitting the model to estimate new volumes, and finally computing an uncertainty for the process from the set of computed volumes. For a full discussion of the bootstrap see Efron and Tibshirani [21] but we will give a brief description here of how we applied the bootstrap method in the current study.

The object of our application of the bootstrap was to develop volume uncertainty as a function of grid size. The process began with the conversion of the Euclidean data points to spherical coordinates. An automatic selection of knots was done. These steps were done once for a given data set. It was assumed that the modifications of the data made during the bootstrap would be small and not affect the knot selection. During the first pass of the bootstrap algorithm the radii of the spherical coordinates were fit by a tensor product of B-splines. The predicted values and residuals from this initial fit were called the master predicted values and master residuals respectively. The computed volume was then put in a list of volumes that was added to in subsequent passes of the algorithm. The algorithm then iterated two hundred times as follows. In the first iteration the master residuals were sampled randomly uniformly with replacement. The resampled residuals were then added to the master predicted radii and a new fit was performed. The new computed volume was added to the volume list and the master residuals sampled randomly uniformly with replacement for the next iteration. The process continued for all two hundred iterations. The standard deviation of the volumes in the volume list was then used as an estimate of the volume uncertainty and the average volume was taken as the reference volume for the chosen grid size. We found that in all cases the expanded uncertainties remained approximately constant and that is why only upper bounds for the expanded uncertainties are reported in the tables below.

Table 6. Estimated Volumes for Calibrated Sphere Data

Grid Case	θ	Calibrate Sphere Data. Volumes vs Grid Size		
		Expanded Volume Uncertainties: Coarse < 0.6 (mm ³), Dense < 0.5 (mm ³)		
		ϕ	Coarse Volume (mm ³)	Dense Volume (mm ³)
1	10	20	3455.05	3453.90
2	20	40	3587.92	3586.74
3	30	60	3610.80	3609.61
4	40	80	3618.63	3617.35
5	50	100	3622.17	3620.88
6	60	120	3624.07	3622.83
7	70	140	3625.21	3623.92
8	80	160	3625.95	3624.66
9	90	180	3626.45	3625.23
10	100	200	3626.81	3625.59
11	110	220	3627.08	3625.86
12	120	240	3627.28	3626.06
13	130	260	3627.43	3626.21
14	140	280	3627.56	3626.34
15	150	300	3627.66	3626.44
16	160	320	3627.74	3626.52
17	170	340	3627.81	3626.59
18	180	360	3627.86	3626.64
19	190	380	3627.91	3626.69
20	200	400	3627.95	3626.73
21	210	420	3627.99	3626.77

Table 7. Estimated Volumes for Green Phantom Data

Grid Case	θ	Green Phantom Data. Volumes vs Grid Size		
		Expanded Volume Uncertainties: Coarse < 12 (mm ³), Dense < 8 (mm ³)		
		ϕ	Coarse Volume (mm ³)	Dense Volume (mm ³)
1	10	20	4125.34	4092.77
2	20	40	4283.92	4250.85
3	30	60	4310.93	4277.65
4	40	80	4320.67	4285.66
5	50	100	4324.90	4289.86
6	60	120	4327.17	4292.12
7	70	140	4328.53	4293.47
8	80	160	4329.41	4294.35
9	90	180	4330.01	4294.95
10	100	200	4330.44	4295.37
11	110	220	4330.75	4295.69
12	120	240	4330.99	4295.92
13	130	260	4331.07	4296.11
14	140	280	4331.33	4296.26
15	150	300	4331.45	4296.38
16	160	320	4331.54	4296.47
17	170	340	4331.62	4296.55
18	180	360	4331.69	4296.62
19	190	380	4331.75	4296.68
20	200	400	4331.80	4296.72
21	210	420	4331.84	4296.77

Table 8. Estimated Volumes for Pink Phantom Data

Pink Phantom Data. Volumes vs Grid Size				
Expanded Volume Uncertainties: Coarse < 28 (mm ³), Dense < 18 (mm ³)				
Grid Case	θ	ϕ	Coarse Volume (mm ³)	Dense Volume (mm ³)
1	10	20	3668.27	3719.71
2	20	40	3811.59	3863.30
3	30	60	3836.22	3887.99
4	40	80	3844.59	3896.39
5	50	100	3848.41	3900.21
6	60	120	3850.46	3902.27
7	70	140	3851.69	3903.51
8	80	160	3852.48	3904.30
9	90	180	3853.03	3904.85
10	100	200	3853.41	3905.23
11	110	220	3853.70	3905.52
12	120	240	3853.91	3905.74
13	130	260	3854.08	3905.91
14	140	280	3854.22	3906.04
15	150	300	3854.32	3906.15
16	160	320	3854.41	3906.24
17	170	340	3854.48	3906.31
18	180	360	3854.55	3906.37
19	190	380	3854.60	3906.42
20	200	400	3854.64	3906.47
21	210	420	3854.68	3906.50

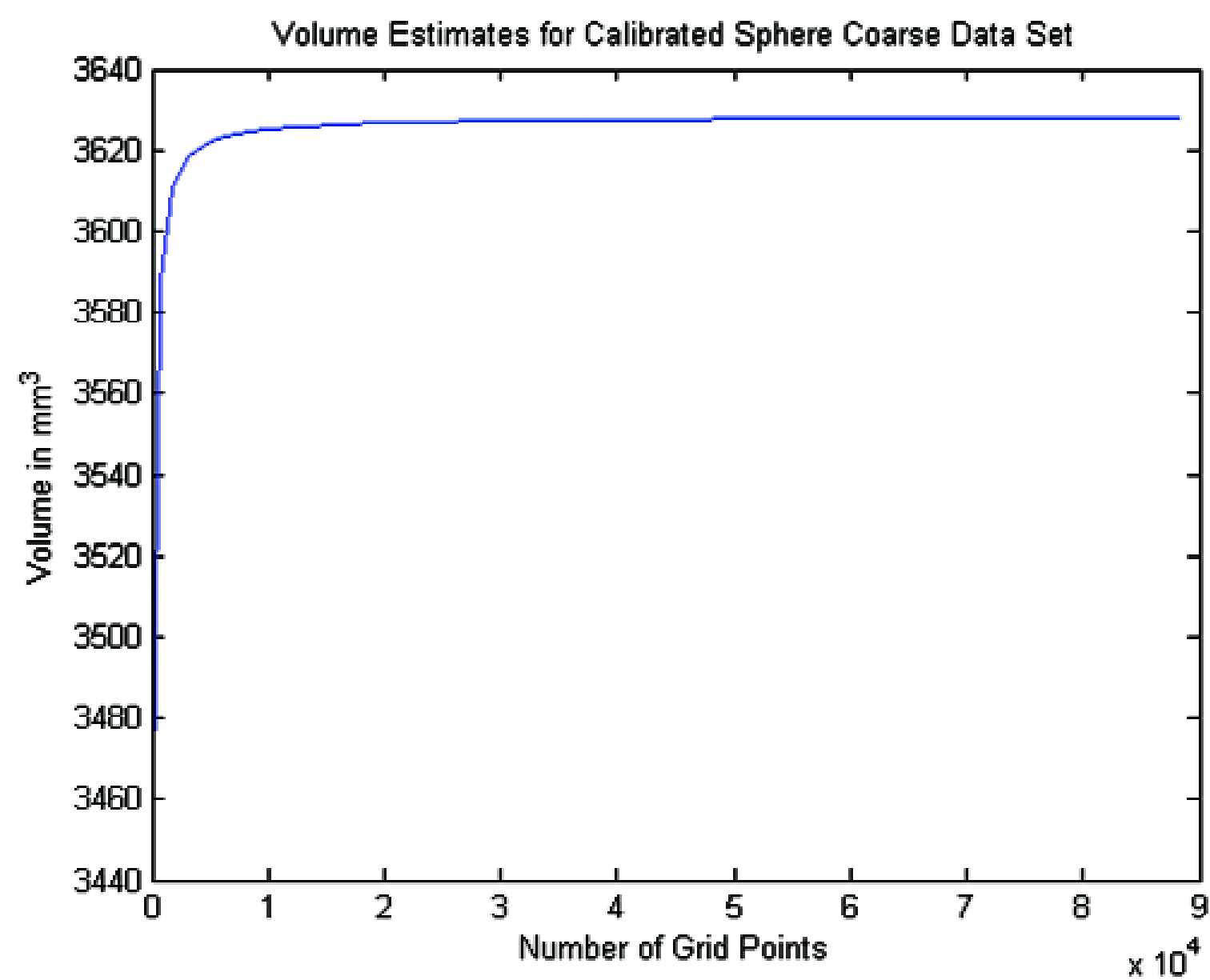


Fig. 14. Volume Estimates for Coarse Calibrated Sphere Data Set vs. Grid Size.

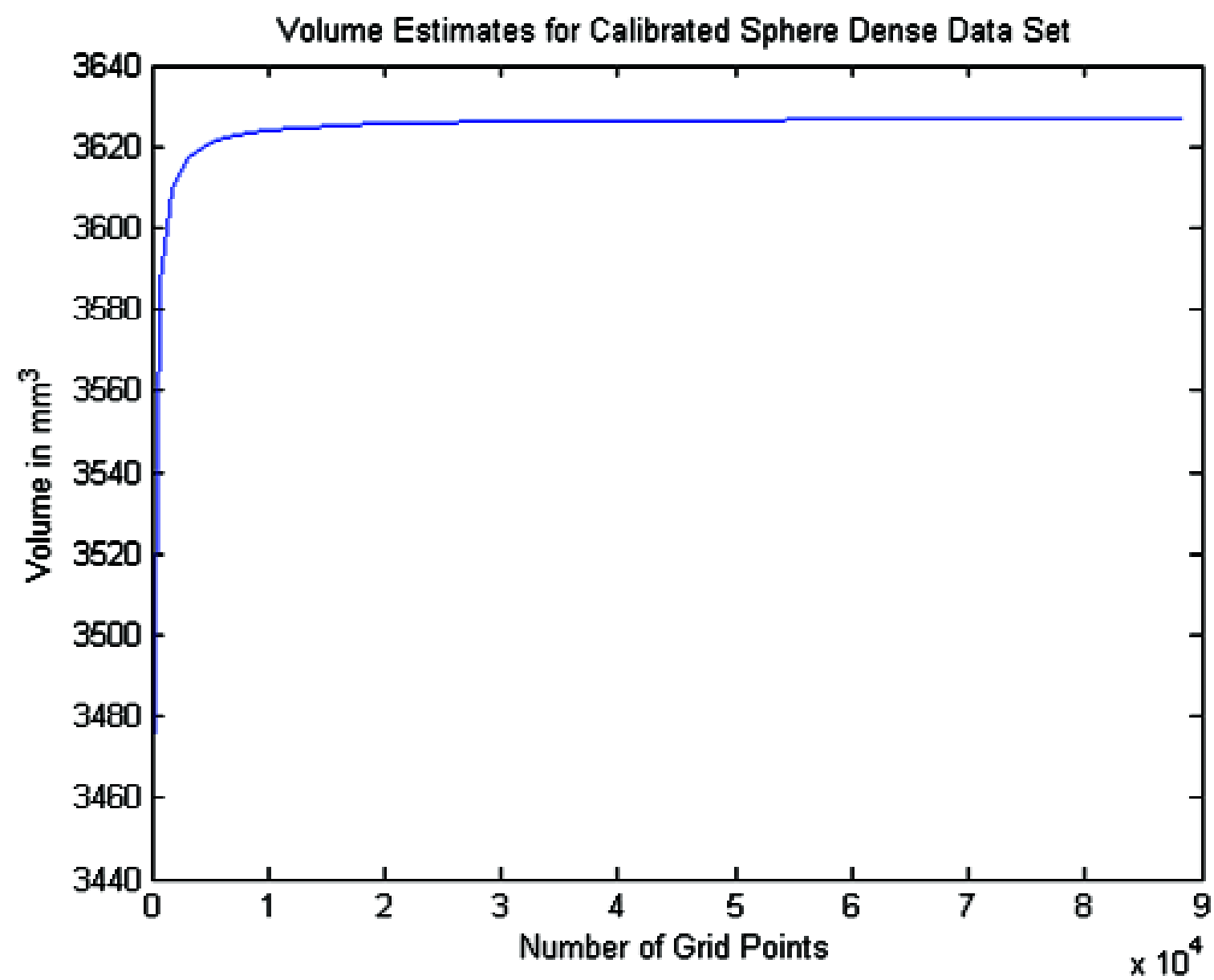


Fig. 15. Volume Estimates for Dense Calibrated Sphere Data Set vs. Grid Size.

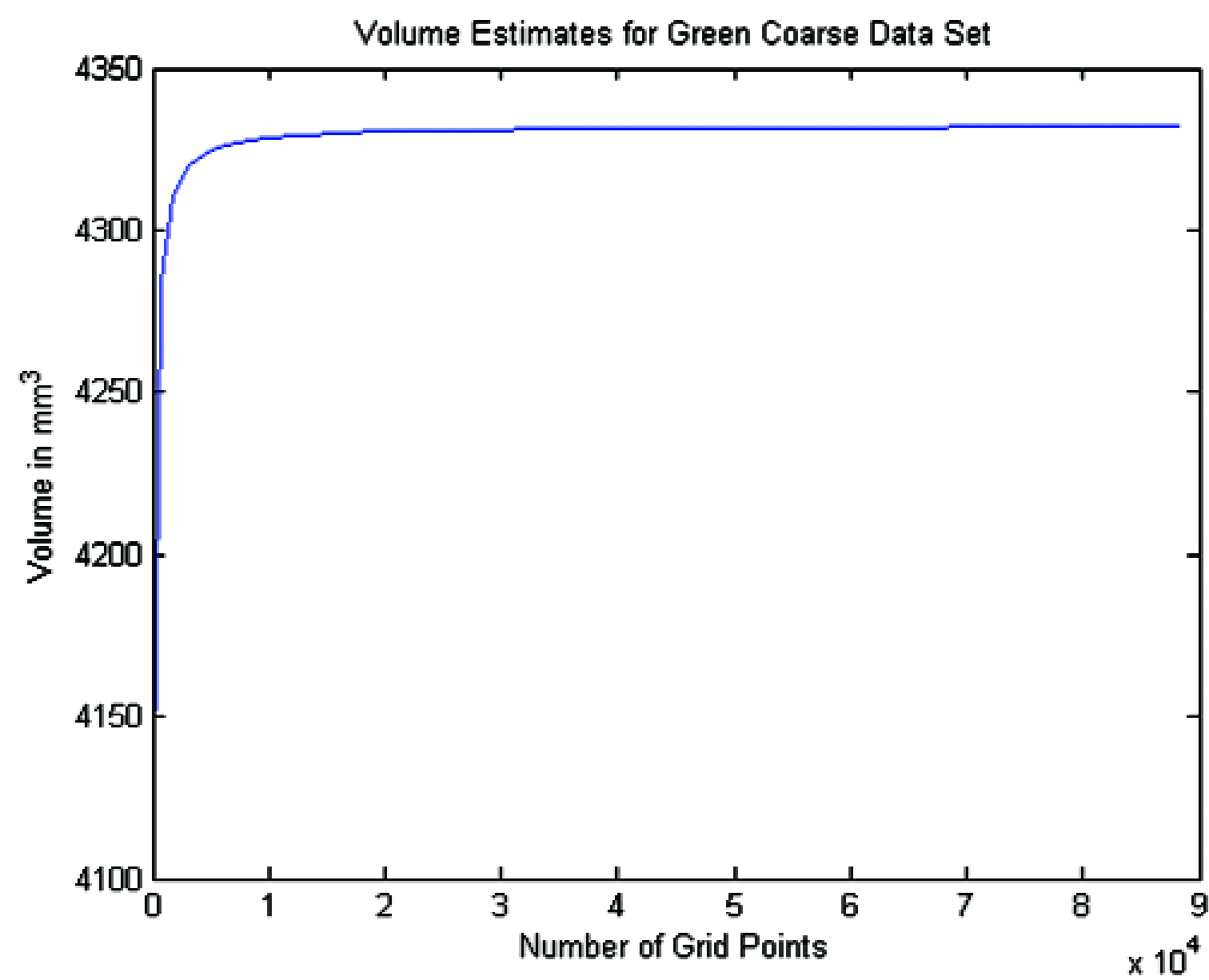


Fig. 16. Volume Estimates for Green Coarse Data Set vs. Grid Size.

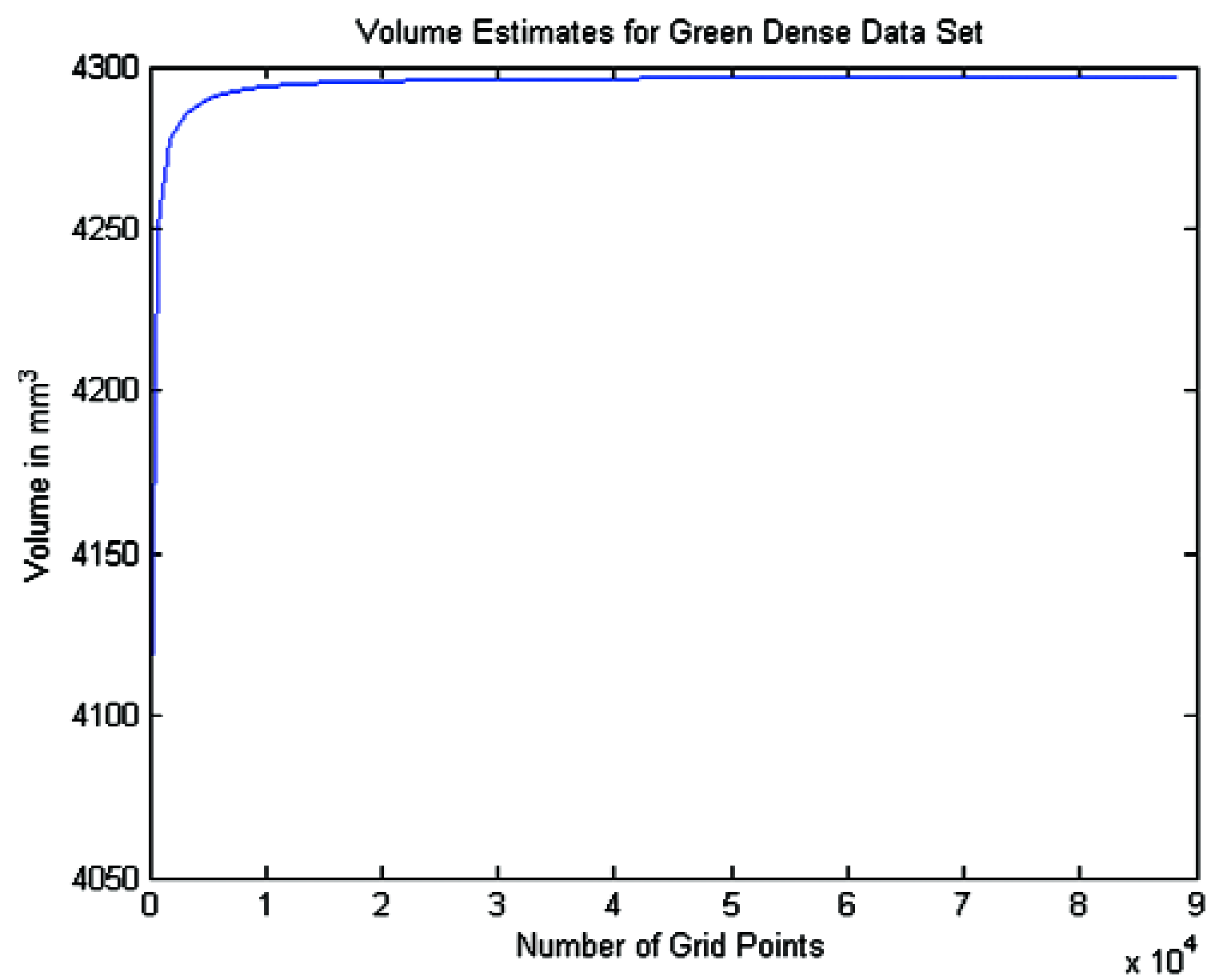


Fig. 17. Volume Estimates for Green Dense Data Set vs. Grid Size.

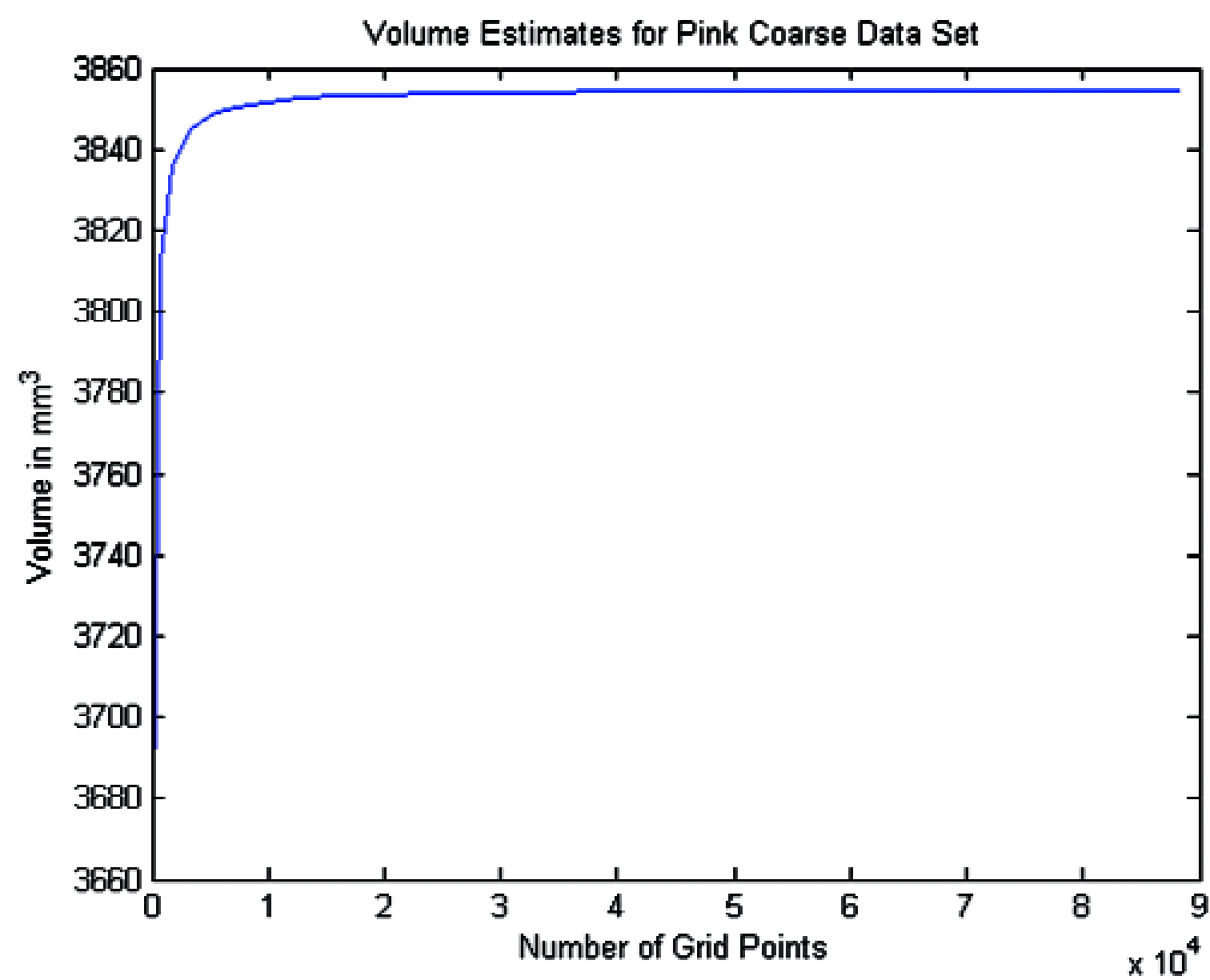


Fig. 18. Volume Estimates for Pink Coarse Data Set vs. Grid Size.

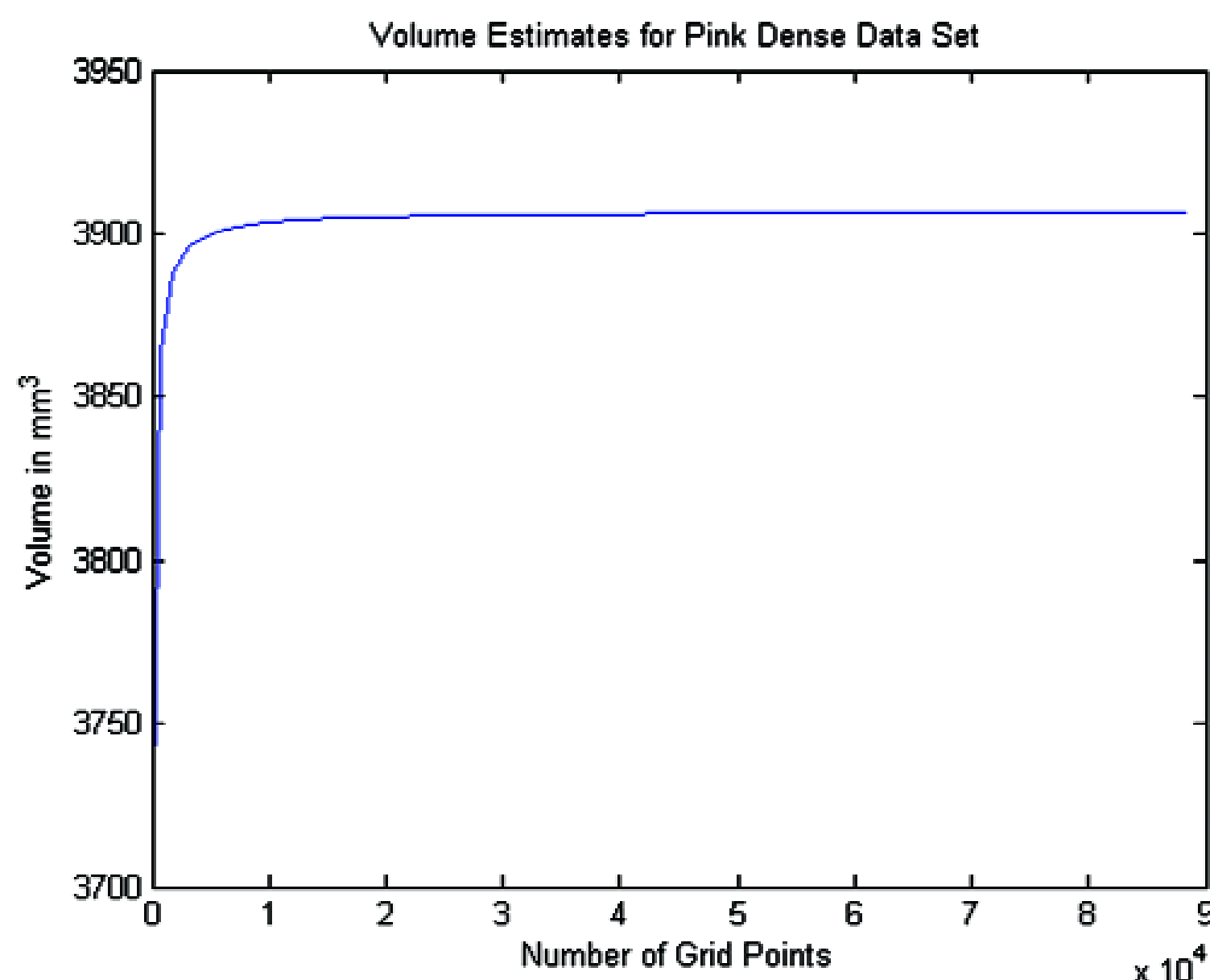


Fig. 19. Volume Estimates for Pink Dense Data Set vs. Grid Size.

6.2.2 Calibrated Sphere Volume Estimates and Uncertainties for the Coarse and Dense Data

As a test of the accuracy of the B-spline volume estimation procedure, we applied the method to the coarse and dense data sets for the calibrated sphere. If the volumes computed by the B-spline method in Table 6 are compared with the sphere fit results in Table 3, we see that the B-spline method and the sphere fit method results differ by approximately 0.01 % for both the coarse and dense data sets. This indicates that the B-spline approach is performing as adequately as the straightforward sphere fit model and thus can be relied upon to produce good results when applied to the phantom data.

6.2.3 Phantom Volume Estimates and Uncertainties for the Coarse Data

In this section we discuss briefly the results of computing the volumes and volume uncertainties for the two phantoms using the coarse data sets. As described earlier, twenty one surface grid cases were used and the results are shown in Tables 7 and 8. Figures 16 and 18 show the trends of the volume estimates as the grid sizes increase. The estimates rise rapidly, and both tend to what appears to be stabilized values. The figures graphically display the values in the tables.

The spherical fit to the Green phantom data in Table 4 indicates an estimated volume of 4331.3 mm^3 . Table 7 indicates an estimated volume of 4331.84 mm^3 for the largest grid size of 210θ by 420ϕ . This is approximately a 0.01 % difference suggesting that the Green phantom is very nearly spherical.

The spherical fit to the Pink phantom data in Table 5 indicates an estimated volume of 3862.6 mm^3 . Table 8 indicates an estimated volume of 3854.68 mm^3 for the largest grid size. The difference is approximately 0.21 %, suggesting the slight non-spherical shape of the Pink phantom.

From Tables 7 and 8, the uncertainties are shown bounded on the coarse surface grid by 12 mm^3 for the Green phantom and 28 mm^3 for the Pink phantom. This larger uncertainty for the Pink phantom might be due to the non-spherical nature of the Pink phantom. As indicated here, the uncertainties for the Pink phantom are approximately twice those of the Green phantom.

6.2.4 Volume Estimates and Uncertainties for the Dense Data

In this section we report the results of volume estimates and the extended uncertainties for the dense data sets for both phantom surfaces. The first thing of interest is that the stabilized volume estimates for the Green dense data in Table 7 differ from the volume estimate

for the Green coarse data by about 0.82 %. This is in the negative direction in that the volume estimates for the Green dense data were smaller than for the Green coarse data. In the case of the volume estimates for the Pink dense data in Table 8, the differences with the volume estimate for the Pink coarse data is approximately 1.33 %. This is in the positive direction in the sense that the volume estimate for the dense data is slightly larger than the volume estimate for the coarse data. These results are consistent with the results of the spherical model fit to the Green and Pink phantoms in Tables 4 and 5. Another thing to notice is that the uncertainties are lower for the dense data sets as shown in Tables 7 through 8 compared to those estimated from the coarse data, although the uncertainty for the Pink phantom is about twice that for the Green phantom. These uncertainties are consistent with the uncertainties computed during the fits of the sphere models in Tables 4 and 5. It is not clear why the results for the dense data differ slightly in the directions they do from the results for the coarse data. A complete analysis of these differences is beyond the scope of this paper, but would be appropriate for a detailed study related to an analysis of the fitting algorithms and their sensitivity to the surface data distribution.

7. Summary

The B-spline surface model joined with the Divergence Theorem seems to be a viable approach for estimating volumes of the near-spherical molded phantoms, but the volume results seem to depend strongly on the distribution of the data points on the surface. In the case of sparse data the problem of extending a surface fitted to the data to a grid on the surface leads potentially to unwanted oscillations in the neighborhood of some of the sparse data points. In the current algorithm this problem was dealt with by adding regularization terms to the objective function. These provided a balance between the fit and the surface smoothness in order to control overfitting of the data. The final regularization parameter was $\lambda = 0.32$ for all data sets.

The results obtained from both the coarse and dense data distributions appear to be consistent with the results from the spherical model fits. As shown in Table 7 the volume estimates for the Green phantom are very close to the estimates obtained by the spherical model in Table 4 suggesting that the Green sphere is essentially spherical in shape. The volume estimates in Table 5 for the Pink sphere are higher by about

0.19 % for the coarse data and 0.15 % for the dense data than the B-spline model estimates in Table 8 suggesting that the Pink phantom might be close to spherical, although it does show some non-spherical tendencies. The uncertainties in the Pink phantom case suggest though that the Pink phantom may indeed have a slightly non-spherical shape. This is also suggested by the difficulty with holding the Pink phantom in the vacuum chuck during one of the surface data probing experiments.

In summary we conclude that the combination of using the CMM to obtain surface data and the B-spline/Divergence Theorem volume estimation method can produce useful results but that there are some limitations. First, the CMM is primarily used for probing manufactured metallic artifacts and its use in probing non-metallic artifacts such as the FDA phantoms can likely lead to some larger uncertainties in volume estimation. Second, the distribution of probed points can lead to results that indicate the models used are sensitive to the point locations. Third, the methods employed may not provide useful volume estimation for more complex artifacts that surely would arise in developing simulated lung cancer phantoms. The current artifacts were near-spherical and even these did not provide fully expected results based on data distribution for the Pink phantom. Further research on the affect of surface data distribution is required. Fourth, the Pole Problem is definitely a limitation that required regularizing of the objective function. Finally, the number of data points obtained on the surfaces was limited by the relative size of the artifact and the CMM probe size.

Acknowledgements

The authors wish to acknowledge the generous assistance of Stefan Leigh of the Statistical Engineering Division of ITL at NIST in designing the volume uncertainty estimation algorithm by the bootstrap method. We also wish to acknowledge the gracious assistance of Prof. Dianne P. O'Leary, University of Maryland, in writing an efficient version of the least squares program using SVD as well as consulting on the regularization of the objective function. She also provided many helpful editorial suggestions. Prof. Florian Potra of the University of Maryland, Baltimore County also provided suggestions on the form of the regularization terms for the objective function. We also thank Dr. Andrew Rukhin of the Statistical Engineering Division of NIST for his many helpful suggestions on the manuscript. We acknowledge the use of the Nelder-Mead MATLAB

code by Prof. Timothy Sauer, George Mason University. Finally, we gratefully acknowledge Dr. Nicholas Petrick of FDA for the loan of the Green and Pink phantoms used in this study.

Disclaimer

Certain commercial software products are identified in this paper in order to adequately specify the computational procedures. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology nor does it imply that the software products identified are necessarily the best available for the purpose.

8. References

- [1] American Cancer Society, Cancer Facts & Figures 2007. Atlanta: American Cancer Society, 2007.
- [2] A. E. Taylor, Advanced Calculus, Ginn and Company, Boston, 1955.
- [3] P. Dierckx, Curve and Surface Fitting with Splines, Clarendon Press, Oxford, 2006.
- [4] P. Dierckx, An Algorithm for Surface-Fitting with Spline Functions, IMA Journal of Numerical Analysis **1**, (1981), 267-283.
- [5] P. Dierckx, Algorithms for Smoothing Data on the Sphere with Tensor Product Splines, Computing **32**, (1984), 319-342.
- [6] P. J. Schneider and D. H. Eberly, Geometric Tools for Computer Graphics, Morgan Kaufmann Publishers, Amsterdam, 2003.
- [7] B. N. Taylor and C. E. Kuyatt, Guidelines for Evaluating and Expressing the Uncertainty of NIST Measurement Results, NIST Technical Note 1297, 1994 Edition, National Institute of Standards and Technology, 1994.
- [8] N. Draper and H. Smith, Applied Regression Analysis, John Wiley & Sons, New York, 1981.
- [9] C. M. Shakarji, Least-Squares Fitting Algorithms of the NIST Algorithm Testing System, J. Res. Natl. Inst. Stand. Technol. **103**, (1998), 633-641.
- [10] T. Sauer, Numerical Analysis, Pearson, Boston, 2006.
- [11] C. de Boor, A Practical Guide to Splines, Springer-Verlag, New York, 1978.
- [12] M. G. Cox, The Numerical Evaluation of B-spline', J. Inst. Maths Applies. **10** (1972) 134-149.
- [13] D. H. Eberly, Game Physics, Morgan Kaufmann Publishers, Amsterdam, 2004.
- [14] D. F. Rogers and J. A. Adams, Mathematical Elements for Computer Graphics, McGraw-Hill, Boston, 1990.
- [15] G. H. Golub and C. F. Van Loan, Matrix Computations, Third Edition, Johns Hopkins University Press, Baltimore, 1996.
- [16] P. C. Hansen, Rank-Deficient and Discrete Ill-Posed Problems, SIAM, Philadelphia, 1998.
- [17] P. C. Hansen, J. G. Nagy, and D. P. O'Leary, Deblurring Images: Matrices, Spectra, and Filtering, SIAM, Philadelphia, 2006.
- [18] M. A. Kilmer and D. P. O'Leary, Choosing Regularization Parameters in Iterative Methods for Ill-Posed Problems, SIAM J. Matrix Anal. Appl. **22**, 4 (2001), 1204-1221.
- [19] D. P. O'Leary, Near-Optimal Parameters for Tikhonov and other Regularization Methods, SIAM J. Sci. Comput. **23**, 4, 2001, 1161-1171.
- [20] A. N. Tikhonov and V. Y. Arsenin, Solutions of Ill-Posed Problems, John Wiley & Sons, New York, 1977.
- [21] B. Efron and R. J. Tibshirani, An Introduction to the Bootstrap, Chapman and Hall, New York, 1993.

***About the authors:** David E. Gilsinn is a mathematician with the Mathematical and Computational Sciences Division (MCSD) in the Information Technology laboratory (ITL) of NIST. He received a Ph.D. in mathematics from Georgetown University in 1969 and has been with NBS/NIST since 1970. His work has spanned the range of applied mathematics problems in urban planning, computer graphics, image processing, and numerical controlled machine tool metrology, error compensation, and vibration control. Bruce R. Borchardt received his B.S. in physics from Yale University in 1971. He has worked as a dimensional metrologist, first at NBS and, later, NIST since then. He has worked on CMMs most of his career and is highly experienced in various methods of coordinate metrology. Amelia Tebbe is a senior mathematics major at St. Marys College of Maryland in St. Marys City, MD. She was a SURF (Student Undergraduate Research Fellow) student in MCSD in 2008 and the Statistical Engineering Division (SED) of ITL in 2007. She is interested in Abstract Algebra and the intersection of mathematics and computer science. She contributed to the programming and graphics aspect of this study and plans to pursue a Ph.D. program in Fall 2010.*